

Curso de Introducción a GNU/Linux: Tecnología

David Fernandez Vaamonde davidfv(at)davidfv(dot)net
Francisco Javier Tsao Santín tsao(at)enelparaiso(dot)org

Escuela Técnica Superior de Ingenieros de Caminos, Canales y
Puertos
Noviembre 2005

Índice general

1. Repaso a las distribuciones más importantes	2
1.1. Los primeros tiempos	2
1.2. Primeras distribuciones de software GNU/Linux	2
1.3. Distribuciones derivadas	3
1.3.1. Basadas en .rpm	3
1.3.2. Basadas en .deb	3
1.3.3. Basadas en gentoo	3
1.3.4. Distribuciones “live”	4
1.4. Unixes libres más allá de Linux: BSD	4
2. Taller: Instalación de Debian 3.1 “Sarge”	5
2.1. Instalación básica	5
2.1.1. Arranque de DVD/CD	5
2.1.2. Selección de locales	5
2.1.3. Inicio del instalador	6
2.1.4. Configuración de red	6
2.1.5. Particionamiento de disco	6
2.1.6. Instalación de sistema base y arranque	8
2.1.7. Configuración del software base	8
2.2. Selección de paquetes	9
2.3. Configuración del sistema gráfico	12
2.4. Modems	13
2.5. Impresoras	13
2.6. Scanners	14
2.7. Capturadoras de televisión	14
2.8. Algunas notas sobre portátiles	14
2.9. Instalación de paquetes externos a la distribución	15
2.10. Recetas para solucionar problemas frecuentes	16
2.10.1. X	16
2.10.2. Teclado (en consola virtual)	17
2.10.3. Internacionalización	17
2.10.4. Arranque con varios operativos	17
2.10.5. Instalación en una máquina que no tiene acceso a cd/dvd	18

3. Algunas herramientas y paquetes libres	20
3.1. Servidores gráficos, gestores de ventanas y escritorios	20
3.1.1. Servidores gráficos	20
3.1.2. Gestores de ventanas	20
3.1.3. Escritorios	21
3.2. Red	21
3.3. Ofimática	22
3.4. Gráficos	22
3.5. Multimedia	22
3.6. Soportes digitales	23
3.7. Matemáticas	23
3.8. Ingeniería	23
3.9. Programación	23
4. Dónde descargar software	24
5. Documentación y más ayuda	25
5.1. Documentación	25
5.2. Foros de noticias	25
5.3. Grupos de usuarios	26
6. Taller: Instalación de Mandriva 10.2	27
A. Configuración wireless para udcwifi	28

Capítulo 1

Repaso a las distribuciones más importantes

1.1. Los primeros tiempos

- Paquetes de software GNU (desde 1985) <http://www.gnu.org>
- Kernel Linux (desde 1991) <http://kernel.org>
- Instalación *from scratch* <http://www.linuxfromscratch.org>

1.2. Primeras distribuciones de software GNU/Linux

- Slackware <http://www.slackware.org> considerada la primera distribución de GNU/Linux, iniciativa de Patrick Volkerding; instalación por paquetes .tgz (tar.gz), compleja de actualizar
- Red Hat <http://www.redhat.com> distribución de una de las compañías pioneras que basan su negocio en GNU/Linux, cotiza en Wall Street, mucha fuerza en toda América; actualmente mantiene dos grupos de distribución, RH Enterprise Edition y Fedora; marcaron el standard de instalación por paquetes .rpm
- Debian <http://debian.org> distribución mantenida por voluntarios esparcidos por todo el planeta, convertirse en desarrollador de Debian es casi como hacer una oposición, exigencia de conocimientos técnicos e inglés, proceso de tutela, exámenes... unos dos años actualmente; las versiones *estables* se distancian cada vez más en el tiempo, según aumenta la cantidad de software, a ojos de una empresa puede parecer un problema, aunque el usuario de Debian casi siempre trabaja sobre versiones testing (que siguen siendo muy estables) o inestables; marca el standard de paquetes

.deb, es reconocida su estabilidad de gestión de dependencias, instalación y actualización de software

- Gentoo <http://gentoo.org> proyecto relativamente reciente, el software es distribuido en código fuente que hay que compilar y las actualizaciones son parches al fuente, con la idea de optimizar en la compilación para la propia máquina

1.3. Distribuciones derivadas

1.3.1. Basadas en .rpm

- Suse <http://www.novell.com/es-es/linux/suse> originalmente alemana, la compañía fue absorbida por Novell (y pese a los rumores sobre problemas económicos, es uno de los grupos estratégicos de Novell)
- Mandriva <http://www.mandriva.com> originalmente Mandrake, empresa francesa, que pasó por problemas de suspensión de pagos, actualmente recuperada, absorbe otras empresas de distribución de Linux en diferentes áreas geográficas, como la brasileña Conectiva (Mandrake+Conectiva=Mandriva); objetivo principal es hacer una distribución lo más fácil de manejar posible; liberan una versión *community* y otra *enterprise* alternativamente, cada seis meses
- Turbolinux <http://www.turbolinux.com> muy extendida en extremo oriente, recientes acuerdos con Red Flag Linux y Fujitsu
- Aurox <http://www.aurox.org> Distribución polaca enfocada a portátiles

1.3.2. Basadas en .deb

- Linspire <http://www.linspire.com> distribución enfocada al mundo empresarial, fácil de usar
- Ubuntu <http://www.ubuntu.com> iniciativa del *afonauta* Mark Shuttleworth, desarrollador de Debian y dueño de Canonical ltd., su objetivo es tener una distribución estable cada seis meses partiendo de la versión inestable de Debian; no hay una versión *empresarial* de distribución bajo pago y otra de libre distribución (*community*) como en otras, hay una única versión libre que además se puede solicitar (en lotes cuanto mayores mejor) sin gastos de envío (<http://shipit.ubuntu.com>); dos series, la original basada en el escritorio Gnome, y otra (kubuntu) basada en KDE.

1.3.3. Basadas en gentoo

- Ututo <http://ututo.org> recomendada por RMS por no incluir/enlazar ningún paquete de software privativo

1.3.4. Distribuciones “live”

- Knoppix <http://www.knoppix.org> impulsada por el germano Klaus Knopper, es el referente de las distribuciones live; basada en Debian
- Ubuntu <http://www.ubuntulinux.org> la versión live de Ubuntu
- Necromantux <http://necromantux.gpul.org> basada originalmente en Knoppix, se cambió luego el método de desarrollo por el de Metadistros (<http://metadistros.software-libre.org>); en versión beta todavía y con instalador en fase de desarrollo, enfocada a recuperación de sistemas, reemplazo rápido de servicios, análisis forense...

1.4. Unixes libres más allá de Linux: BSD

- FreeBSD <http://www.freebsd.org> primer heredero del unix de Berkley, liberado tras una pugna legal, el objetivo principal es la arquitectura i386; probablemente se hubiera impuesto si Linus Torvalds se hubiera retrasado dos o tres años
- NetBSD <http://www.netbsd.org> *fork* del proyecto anterior, su objetivo es funcionar en el mayor número de arquitecturas posibles
- OpenBSD <http://www.openbsd.org> *fork* de NetBSD, su fundador, Theo De Raat emigró a Canada para poder desarrollar la versión más paranoica, con los cifrados más fuertes y la instalación por defecto más segura de todos los unixes; de este proyecto derivan otros tan importantes y tan portados como OpenSSH; no apto para novatos
- OpenSolaris <http://www.opensolaris.org> Sun Microsystems en su nueva estrategia de *open source* (más que de *free software*) ha empezado a liberar software para construir una versión libre de su conocido Solaris
- Schillix <http://schillix.berlios.de> un live cd basado en el primer software solaris liberado por Sun

Más información sobre distribuciones: <http://www.distrowatch.org>

Es importante escoger una distribución con cierta idea, y *trabajársela* personalizándola. No es nada recomendable saltar de distribución en distribución hasta encontrar alguna que funcione todo, así sólo se pierde tiempo y se retrasan los problemas que habrá que afrontar.

Capítulo 2

Taller: Instalación de Debian 3.1 “Sarge”

2.1. Instalación básica

Antes de proceder a la instalación, se recomiendan dos cosas (que rara vez se cumplen):

- Sacar copias de seguridad de los datos que se quieran conservar en la máquina
- Leer la documentación de la distribución. En el caso de Debian, además de en la web, en los propios cd's/dvd's

2.1.1. Arranque de DVD/CD

- El arranque por defecto se realiza con un kernel de la serie 2.4
- Si se requiere un kernel más moderno (véase mejor soporte para discos SATA), o distintas opciones de arranque por hardware más o menos extraño, pulsando F1 está la información para las distintas opciones, v.g.”linux26”, para arrancar con esa serie de kernels.
- Se puede escoger también antes de arrancar un instalación en modo experto.

2.1.2. Selección de locales

- Se selecciona idioma
- Se selecciona país
- Se selecciona tipo de teclado

2.1.3. Inicio del instalador

- Detección automática de dispositivos de cdrom
- Análisis del cdrom/dvdrom
- Carga módulos del instalador

2.1.4. Configuración de red

- Detección del hardware de red
- Si hay dispositivos de red, intenta encontrar un servidor DHCP (asignación automática de IP)
- Si falla la conexión por DHCP, entonces se puede configurar manualmente (pero no es necesario si tenemos la distribución en el dvd):
 - Se introduce la IP de la máquina
 - Se introduce la máscara de red
 - Se introduce la IP de la puerta de enlace
 - Se introducen las IP's de servidores de DNS
- Se introduce el nombre de la máquina (se puede cambiar después editando `/etc/hostname`)
- Se introduce el dominio en el que está la máquina, si se ha configurado la red

2.1.5. Particionamiento de disco

Se escoge la unidad en la que se va a instalar el sistema, y si se va a usar completa o no. Si se tiene otro operativo y se quiere conservar es obligado haber hecho una reducción previa de su partición. Si ese operativo es Windows, se habrá desfragmentado previamente a la reducción (si es otro sistema unix probablemente no sea necesario ya que se suelen usar sistemas de ficheros de muy baja fragmentación), y, según que versiones (95/98/XP), será necesario dejarle el espacio a principio de disco para que pueda arrancar.

Herramientas libres para realizar esta clase de operaciones son **nparted**, **qtparted**, **gparted**.

Además de los sistemas clásicos, también se pueden hacer instalación en RAID por software, y otros modelos más avanzados que se salen del propósito de este taller.

Se presentan varios estilos de particionamiento. Todos tienen en común la necesidad de crear al menos una partición de datos y una de *swap* o intercambio (al contrario que en los sistemas de Microsoft, el espacio de intercambio no se realiza en un fichero sino en una partición aparte). Es recomendable usar del orden del doble de memoria física de la máquina, más si tenemos intención de prepararla para realizar suspensiones a disco.

- Para usuarios que se inician, se recomienda todo el sistema (salvo intercambio) en una partición. Personalmente me parece una decisión muy equivocada.
- Para equipos destinados a ofimática, realiza una separación entre espacio de datos (directorio /home) y espacio de sistema. Es mucho más correcto.
- Finalmente, se puede realizar una partición a medida. El estilo de reparticionamiento para un servidor casero actual (orden de 100 Gb de disco) podría ser 500 Mb para /, 100 Mb para /boot, 10 Gb para /usr, 15 Gb para /var (esto, dependiendo del uso que se le quiera dar, servidores de correo, impresión o similares pueden exigir mucho espacio), 5 Gb para /home (dependiendo del uso que le vaya a dar uno o varios usuarios, si se van a manejar ficheros voluminosos), 5 Gb para /tmp (un espacio interesante para poder crear imágenes de DVD; recordar que /tmp por defecto se borra al reiniciar la máquina); y lo dicho, el doble o más de la memoria física para intercambio.

Por herencia de BIOS antiguas, los discos permiten hasta 4 particiones primarias, cada una de ellas se puede dividir en particiones lógicas. Si no se va a ocupar todo el disco con Debian, es recomendable tomar una partición primaria para /, y a partir de ella, escoger sólo particiones lógicas. En cualquier caso, el sistema de instalación avisa en el momento en que un espacio libre quede inutilizable por el particionamiento.

Para definir las particiones:

- Se puede seleccionar espacio libre y entonces:
 - Se indica el tamaño de la nueva partición (por defecto toma todo el espacio libre)
 - Se decide el tipo de partición (primaria/lógica)
 - Se señala para situar al principio o al final del disco
 - Menú de definición de la partición:
 - Utilizar como: ext3, reiserfs, jfs, xfs... El sistema de ficheros nativo de Linux es ext3, evolución *journalized* de ext2, tiene fama por su estabilidad; otros dan preferencias a otras características, como reiserfs, que prefiere la rapidez. También se puede (y se ha de hacer en alguna partición) seleccionar *swap* o espacio de intercambio: al contrario que en otros sistemas, la memoria virtual se extiende sobre una partición y no sobre un archivo. Eso puede restar flexibilidad en algunos aspectos, pero da más rapidez y estabilidad.
 - Punto de montaje: directorio en el que se va a montar la partición
 - Marca de arranque: indicar si la partición va a llevar un gestor de arranque o no

- o Opciones de montaje: en general, seleccionar opción por defecto, para casos más específicos se puede indicar sólo lectura, imposibilidad de ejecución...
- Se puede seleccionar una partición existente, el menú será similar salvo que habrá una opción para eliminar la partición (y volver al menú anterior) o mantener datos, con lo que se mantendrá la información existente. Esta última opción es interesante en particiones sólo de datos, y vigilando lo que sucede con los ficheros de configuración que puedan existir en ellas (como los ficheros de configuración de usuarios en /home).

2.1.6. Instalación de sistema base y arranque

Los últimos pasos son la instalación del sistema base (que se hace automáticamente). En ese momento también se puede cambiar, desde el menú principal, el nivel de preguntas durante la configuración de los paquetes (sistema **debconf**).

Respecto al sistema de arranque, por defecto se instala **GRUB**, del proyecto GNU. En versiones anteriores de Debian se utilizaba **LILO**, y aún otras distribuciones importantes siguen utilizándolo, pero quizás GRUB sea más completo.

Se hace una autodetección de otros sistemas instalados. Si hay otros sistemas se incluirán en el menú de arranque, que se puede modificar en /boot/grub/menu.lst o /etc/lilo.conf. En el primer caso, basta modificar el fichero de arranque, en el segundo, habrá que ejecutar lilo después de los cambios.

El indicar si el gestor de arranque se instala en el sector de arranque del disco o de la partición se decide según que otros operativos tengamos. Caso de Windows, es recomendable colocar el gestor en el sector de arranque de disco, ya que es bastante más complicado desde el gestor de arranque de Windows hacer arrancar Linux.

Si se ha de instalar un sistema Windows de la serie 95/98/XP habrá que tener en cuenta que instalará el gestor de arranque al principio de disco sin contemplaciones, por lo que habrá que rearrancar el sistema GNU/Linux con un disco de arranque (habrá que utilizar paquetes como **grub-disk**, **yard** o similar para generarlo). Antiguamente, en el proceso de instalación se daba esta opción, hoy se ha abandonado probablemente por la proliferación de live-cd's. Más adelante veremos como solventar estos problemas.

El sistema se reinicia.

2.1.7. Configuración del software base

Después de rearrancar el sistema, se produce la configuración del sistema base:

- Reloj fijado a GMT o no
- Zona horaria
- Clave de superusuario (dos veces, para confirmar)

- Creación de un nuevo usuario: login, nombre completo, clave de usuario (dos veces)
- Configuración del sistema apt: selección de la fuente de paquetes:
 - cdrom (se hace un scanner de cada cd/dvd con paquetes)
 - http
 - ftp (selección de país y mirror)
 - local (repositorio en una partición montada en el sistema o por nfs-network file system)
 - edición manual
- Selección de paquetes por perfil de usuario: estación ofimática, servidor web, servidor DNS... o selección manual

2.2. Selección de paquetes

El sistema de manejo de paquetes Debian tiene cuatro capas:

- Sistemas *gruesos* de selección de paquetes: son el comentado ya de **selección por perfiles**, también el programa **tasksel**, que selecciona grupos de paquetes por tareas: programación, X, TeX... Estos sistemas son interesantes para la persona que no está familiarizada con la paquetería de programas para GNU/Linux. Una buena idea es seleccionar el perfil o task que más se ajuste, y con el tiempo, ir eliminando paquetes que no se usan y añadiendo otros que sí se usan
- Sistemas *refinados* de selección de paquetes: **dselect** (obsoleto), **aptitude** (que surge de la opción *selección manual* del menú de perfiles al final de la configuración del sistema base, o se puede ejecutar desde consola), **synaptic** (para ejecutar en modo gráfico). Todos son front-ends del sistema apt, y es en este nivel donde empezamos a ver las calificaciones y clasificaciones de paquetes:
 - Por estado: instalados o no, actualizables, nuevos, obsoletos o creados localmente...
 - Por características: administración, sistema base, documentación, editores, gráficos, matemáticas, bibliotecas, red...
 - Por prioridad: standard, recomendado, opcional,...

Cada programa consulta la base de datos del sistema de paquetes, y, utilizando por debajo apt, instala/actualiza/elimina y gestiona las dependencias de cada situación.

- El sistema **apt**: la perla de la distribución Debian, gestiona la instalación, actualización y eliminación de paquetes y sus dependencias. Los comandos del sistema permiten cambiar las fuentes de paquetes, búsqueda y muestra de información de cada paquete, resolución de paquetes rotos...
- **dpkg**: la herramienta base del sistema de paquetes Debian. Permite tratar paquetes individualmente, pero no gestiona dependencias, sólo avisa de la no posibilidad de instalar o eliminar el paquete indicado porque dependa o sea dependencia, respectivamente, de otros.

Aptitude y Synaptic son muy fáciles de manejar, se selecciona un paquete, el programa muestra información sobre él y se indica instalar/mantener/borrar según la necesidad, y el programa responde con la información de los paquetes que se instalarán o eliminarán por causa de las dependencias.

Lo realmente interesante está en los comandos básicos del sistema apt:

- Fuentes de paquetes:

- edición manual del fichero `/etc/apt/sources.list`:

```
deb http://ftp.fi.debian.org/debian sarge main non-free contrib
deb http://ftp.fi.debian.org/debian-non-US testing/non-US main non-free contrib
deb http://security.debian.org/ sarge/updates main contrib non-free
```

con lo que estamos eligiendo:

- el repositorio de paquetes (`http/ftp...`)
 - el nivel de estabilidad de los paquetes, en genérico (`stable`, `testing`, `unstable`) o en clave (a noviembre de 2005: `sarge=stable`, `etch=testing` `sid=unstable`)
 - clasificación de paquetes por licencia (`main`, `contrib`, `non-free`)
 - `apt-cdrom add`: permite analizar un cdrom/dvdrom, cambia el `sources.list` (y no se debe editar a mano después), y también actualiza la lista de paquetes
- Actualización del listado de paquetes:
 - `apt-get update` (contra servidor)
 - `apt-cdrom add`
 - Actualización de paquetes:
 - `apt-get upgrade`
 - `apt-get dist-upgrade` (para actualizaciones importantes, entre dos versiones estables, entre estable y testing...)
 - Búsqueda, instalación y eliminación de paquetes:
 - `apt-cache search palabra(s) clave(s)`: busca paquetes por descripción

- apt-cache show nombre_paquete: muestra información sobre paquete (versión, estado, mantenedor, descripción...)
- apt-get install nombre_paquete: instala paquete, sus dependencias y muestra paquetes recomendados para ese paquete
- apt-get remove nombre_paquete: elimina paquete y los que dependen de él
- Solución de problemas:
 - apt-get install -f: instala paquetes a medio instalar, sus dependencias, etc...
 - apt-get check: informa sobre posibles problemas de paquetes

Para complementar el manejo de apt, es interesante conocer un poco el manejo de dpkg:

- dpkg -i nombre_paquete.deb: instala paquete
- dpkg -r nombre_paquete: elimina paquete
- dpkg --purge nombre_paquete: elimina paquete y ficheros de configuración
- dpkg -l: lista todos los paquetes instalados
- dpkg -s nombre_paquete: informa sobre el estado de un paquete
- dpkg -c nombre_paquete.deb: lista los ficheros que contiene un paquete
- dpkg -S nombre_fichero: indica en que paquete instalado está un fichero
- dpkg-reconfigure nombre_paquete: reconfigura el paquete

En el proceso de instalación de paquetes, entra en funcionamiento el sistema de configuración de paquetes Debian, debconf. Hay diferentes formas de que debconf presente sus pantallas de configuración, incluso en modo gráfico, pero es usual utilizar la versión en modo texto Dialog; también se puede establecer (de hecho se puede cambiar en la primera fase de la instalación, como se ha dicho) una gradación de preguntas según el perfil del usuario, desde novato hasta experto. Para cambiar estos aspectos, se puede ejecutar: dpkg-reconfigure debconf

Es muy interesante conocer el hardware sobre el que se trabaja. Actualmente tanto Debian como otras distribuciones, tienen sistemas de autodetección de hardware. Aún así puede haber momentos en que esta detección falle. Herramientas útiles para saber que hardware tenemos son los comandos **lspci** y **lsusb**; una live-cd también ayuda mucho.

A la hora de actualizar el sistema, es recomendable frenar el mayor número de servicios posibles, incluyendo el sistema gráfico. En principio, en el proceso de actualización los servicios se deberían apagar y encender automáticamente, pero nunca está de más evitar riesgos.

2.3. Configuración del sistema gráfico

Al contrario que en sistemas donde kernel y capa gráfica están indisolublemente unidos, los sistemas Unix están perfectamente separados, lo que supone que el sistema mejora su rendimiento, ya que es algo que no necesariamente tiene que estar funcionando; es más estable, ya que la caída del sistema gráfico no implica la caída de todo el sistema; y es más flexible, puesto que la estratificación del sistema gráfico permite variedad de herramientas (ver sección 3.1).

Los sistemas X tienen la típica arquitectura cliente-servidor, y por lo tanto, hay paquetes para unos y para otros, pero existe un metapaquete que permite instalar el sistema gráfico completo si no se conoce a fondo: `x-window-system`. La configuración puede ser tan fácil como que autodetecte todo el hardware sin problemas, o tan complejo como tener que saber hasta el tipo de conversor analógico-digital de la tarjeta gráfica, si no está bien soportada.

Habrá que tener en cuenta estos aspectos (que es posible que autodetecte y recomiende el sistema de configuración `debconf`):

- Teclado: `rules` (`xfree86`, por defecto suele funcionar), `teclado`(`pc101`, `pc102`, `pc105`... según número de teclas), `layout`(es para teclados en español)
- Ratón: `/dev/psaux` para ratones PS/2 y touchpads, `/dev/input/mice` para ratones usb
- Monitor: frecuencias horizontal, vertical... consultar especificaciones del monitor
- Tarjeta de video: según fabricante; si todo falla, se prueba el controlador `vesa`
- Resolución y profundidad de colores: también depende del monitor: 800x600, 1024x768... y 24 bits para color verdadero

Si durante la configuración/lanzamiento del sistema X falla algo, siempre se podrá saltar a una consola virtual para cortar el proceso de configuración. Además del sistema `debconf`, X tiene sus propias herramientas para configurarse (ver subsección 2.5.1).

El sistema X de por sí es poco usable, hay que instalar como mínimo un gestor de ventanas, que empieza a incluir menús y otras ayudas, hasta escritorios (cuando hay un importante grado de integración entre aplicaciones) (ver de nuevo, sección 3.1); tanto gestores como escritorios pueden requerir configuración, pero ya una vez instalados.

El sistema gráfico se puede lanzar desde una consola virtual, con `startx`, o X, o `xinit`, típicamente, o puede estar funcionando un programa de fondo que espere una autenticación y/o selección de gestor de ventanas/escritorio: los típicos son `xdm` (X), `gdm` (Gnome), `kdm` (KDE). Estando funcionando este programa, se puede acceder a su ventana en lo que sería (en condiciones normales) la consola virtual 7; al arrancar la máquina será la pantalla por defecto.

2.4. Modems

Para modems serie clásicos, se generan los dispositivos `/dev/ttys0`, `ttys1...` Se configura la conexión, por ejemplo con el paquete **pppconfig**. Si el modem es uno de los conocidos como winmodems (modems que tienen el cableado mínimo y el resto lo hacen con emulación en el sistema operativo), clásicamente en los portátiles, uno debe cruzar los dedos e irse a la página del proyecto Linmodems (<http://www.linmodems.org>), descargarse el script de detección del modem, y esperar a que esté documentado y soportado

Los modems ADSL suelen ser otro dolor de cabeza, lo primero es descubrir que chipset lleva (ya que puede haber modems de marcas y modelos diferentes con el mismo chip), y buscar un proyecto que soporte ese chipset. Por poner un ejemplo orientativo: en el caso (desgraciado) que uno tenga conexión con Wanadoo que le hayan proporcionado un modem Sagem, uno descubre que lleva un chip Eagle, existe un proyecto que programa ese driver (<http://www.eagle-usb.org>) que actualmente está en Debian ya (por cierto, que la empresa que fabrica los modems ha abandonado su propio driver y ha adoptado ese). El paquete carga un módulo en el núcleo y el firmware en el modem, la conexión se lanza después de haber sido configurada adecuadamente con el paquete **pppoe** (ppp over ethernet). Los sistemas de configuración de red del panel de control de los escritorios funcionan también con modems más o menos conocidos, es otra opción. En fin, cada uno es un mundo.

Si hemos sido hábiles y tenemos un modem router, o la máquina es una más en una red local:

- Si hay un servidor DHCP (los modem-routers más modernos lo suelen incluir), se puede lanzar un cliente (**dhclient** o **pump**, típicamente), o se
- Se puede asignar la IP estáticamente, con **ifconfig**

Si queremos que la red quede configurada permanentemente, se puede configurar `/etc/network/interfaces` para que lance el cliente dhcp o asigne la red local al arrancar, por ejemplo:

```
iface eth0-home inet static
    address 193.144.10.10
    netmask 255.255.255.0
post-up route add default gw 193.144.1.1
```

donde se está indicando la dirección ip fija (193.144.10.10), máscara de red 255.255.255.0, y una vez configurado (post-up) se lanza el comando route para que indique como puerta de enlace 193.144.1.1

Más opciones en la página man de **interfaces**.

2.5. Impresoras

Al igual que los fabricantes de modems, la otra gran lacra del mundo informático son los fabricantes de impresoras. Como aquellos, muchos fabricantes

han optado por recortar por el cableado para abaratar costes, de manera que tenemos winprinters. Aún así, se ha realizado un importante esfuerzo para soportar la mayoría de las impresoras en el mercado, con controladores y con filtros que traducen los ficheros al lenguaje propio de la impresora. Las impresoras mejor soportadas son las postscript. Para más información, sobre impresoras soportadas y como hacerlas funcionar, la página de referencia es la del proyecto linuxprinting (<http://www.linuxprinting.org>).

Para configurar impresoras, hay distintos sistemas:

- `magicfilter`: casi obsoleto, pero suficiente para muchas impresoras
- `foomatic`: scripts del proyecto linuxprinting, que se integran también en herramientas de configuración de escritorio
- `cups`: muy potente, también se integra en configuración de escritorio, o por web en un puerto local.

Debian tiene su propio paquete de configuración estilo `debconf` que exige tener en el sistema `foomatic` y `cups`, es el paquete **`printconf`**, que autodetecta bastante bien las impresoras y configura de forma muy sencilla una impresora en local.

Además de imprimir desde cada aplicación, hay una serie de comandos de impresión: `lp` (imprime), `lpq` (muestra el estado de la cola de impresión) `lprm` (elimina procesos de impresión) que se pueden ejecutar para imprimir ficheros desde consola, pero habrá que comprobar que la cola pasa el tipo de fichero por el filtro adecuado (eso se debe hacer en la configuración, de forma manual o automática)

2.6. Scanners

La madre de todas las herramientas para manejar scanners: **`sane`**. Originalmente concebida para soportar scanners scsi, hoy maneja todo tipo de scanners. Configuración gráfica.

2.7. Capturadoras de televisión

Se detectan por el kernel y se configuran con programas sintonizadores (**`xawtv`**)

2.8. Algunas notas sobre portátiles

En general, no debiera haber demasiada diferencia entre equipos de sobremesa y portátiles, aunque lo limitado del espacio ha impulsado a los fabricantes a incluir linmodems en vez de modems decentes. Otros aspectos que se pueden tener en cuenta:

- energía: un paquete como `laptop-mode`, configurable, minimiza el gasto de energía, apagando los discos, reduciendo el consumo de la pantalla... otros aspectos configurables son la velocidad del procesador, que se puede modificar según esté funcionando con batería o enchufado, entrando ya en recompilación del kernel
- suspensión a memoria y disco: uno de los aspectos interesantes de los portátiles es la posibilidad de apagar la máquina sin perder el estado de trabajo. APM es el sistema antiguo del kernel, pero que energéticamente funciona mejor en muchos portátiles, pero sólo permite suspensión a memoria; ACPI es el sistema moderno, que permite suspensión a disco y a memoria, aunque en según que casos, no funciona tan bien como apm. Para más información, dirigirse a la documentación de ACPI en el paquete del kernel
- pcmcia: las tarjetas de expansión de comunicaciones típicas son las pcmcia(16 bits)/ cardbus(32 bits), el paquete que gestiona ese hardware es el **pcmcia-cs**
- wifi: aunque no sea una tecnología exclusiva de los portátiles, suelen ser éstos los que más uso hacen de ella, sea como tarjetas minipci o pcmcia. El paquete principal que gestiona estos dispositivos es **wireless-tools**. El soporte se puede hacer bien desde el kernel, bien por módulos que se acoplan a él, según la tarjeta. Una vez funcionando, se pueden hacer conexiones sin y con cifrado WEP (original de la especificación 802.11, en extinción por lo malo que es); para cifrado más fuerte, se utiliza por ejemplo WPA, que tiene un paquete de cliente, **wpa supplicant** dentro de Debian. Más información sobre wireless, en Seattle Wireless (<http://www.seattlewireless.net/>), con unas estupendas bases de datos de hardware soportado, y Madrid Wireless(<http://www.madridwireless.net/>)

Páginas de referencia con enlaces a configuraciones para varias distribuciones y muchos modelos y marcas de portátiles, son las de `tuxmobil` (<http://www.tuxmobil.org>) y `linux laptop` (<http://www.linux-laptop.org>)

2.9. Instalación de paquetes externos a la distribución

Como se ha indicado, para instalar un paquete que no esté incluido en la distribución, si es debian, se utiliza `dpkg -i`. Otros casos habituales:

- Paquetes `.rpm`: se puede utilizar la herramienta **alien**, que convierte el `.rpm` en `.deb`, pero no siempre funciona adecuadamente, puede suceder que haya que modificar ficheros de configuración o scripts; otra alternativa es obtener el código fuente del paquete y modificarlo y recompilarlo para después hacer otro paquete Debian, o hacer lo que sigue

- Paquetes .tar: los paquetes genéricos se pueden debianizar, lo que es complejo pero tiene la ventaja de que se incluyen en la base de datos y es fácil de instalar/actualizar/eliminar; pero en general suelen venir preparados (dependiendo de los desarrolladores siempre) para hacer una instalación casi *from scratch*, esto es, colocan cada fichero ejecutable, script, configuración, en el lugar adecuado, que si se quiere actualizar/eliminar se deberá hacer a mano si no se incluye dentro del propio paquete un modo que lo haga automáticamente. Sistemas que nos solemos encontrar:
 - config.sh+install.sh+uninstall.sh: scripts que detectan las herramientas que van a necesitar en el sistema (compiladores, gestores de módulos...), generan un fichero que recoge el install.sh y actúa en consecuencia; también es posible que haya que pasar determinados parámetros a los scripts
 - Makefile (make config, make install...) más propio de paquetes que hay que recompilar necesariamente. make es una herramienta que, entre otras cosas, permite organizar la compilación, enlazado con bibliotecas, etc, lo que se recoge en uno o varios *Makefiles* donde se indican las órdenes, los prerrequisitos y las opciones

Para cada caso, se debe leer detenidamente el fichero de información de compilación/instalación del paquete.

2.10. Recetas para solucionar problemas frecuentes

2.10.1. X

- Cerramos X, si es posible *suavemente*, si no, Ctrl+Alt+Backspace. Si está funcionando xdm, gdm o kdm (estos programas que hacen ese login gráfico tan bonito), parar previamente ejecutando

```
/etc/init.d/xdm (o gdm, o kdm) stop
```

- Opciones:
 - Editamos


```
/etc/X11/XF86Cofig-4
```

 a mano (si sabemos lo que queremos hacer)
 - Utilizamos `xf86cfg -textmode`, o `xf86config`, para reconfigurar X
 - Ejecutamos `dpkg-reconfigure xserver-xfree86` (si es necesario, subir el nivel de preguntas en `debconf`)
- Relanzamos X ejecutando

```
startx
```

- Si algo sigue fallando, volvemos a empezar
- Si todo va bien, podemos relanzar xdm/gdm/kdm con

```
/etc/init.d/xdm start
```

2.10.2. Teclado (en consola virtual)

Ejecutamos

```
dpkg-reconfigure console-data
```

y seleccionamos el teclado adecuado

2.10.3. Internacionalización

Si hay problemas con el mapa de caracteres (que debería ser ISO-8859-1 o ISO-8859-15 o los más valientes UTF-8), se puede reconfigurar ejecutando:

```
dpkg-reconfigure locales
```

O editando a mano `/etc/locale.gen`. Para cambiar la internacionalización sólo en el usuario, se debe tener instalado el paquete `language-env` y ejecutar (como usuario)

```
set-language-env
```

, cambiará los ficheros de configuración de usuario de cada programa (los *.algo*).

2.10.4. Arranque con varios operativos

Ya se indicó la problemática de instalar un sistema que *pise* el gestor de arranque del sistema GNU/Linux que hayamos instalado. Si no disponemos del diskette de arranque, podremos usar un live-cd, como knoppix o necromantux, el procedimiento sería según los casos:

grub

- Montar la unidad donde esté `/boot`: `mount -o rw /dev/sda1 /mnt/sda1`
- Editar `menu.lst`, si es necesario: `joe /boot/grub/menu.lst`
- Reinstalar grub en el sector de arranque del disco (si es que la live tiene grub): `grub-install --root-directory=/mnt/... /dev/sda`

lilo

- Montamos la partición donde esté `/etc/lilo.conf` `mount -o rw /dev/sda1 /mnt/sda1`
- Editamos (si es preciso) `lilo.conf`: `joe /etc/lilo.conf`
- Lanzamos una consola que toma como raiz el punto de montaje de la partición: `chroot /mnt/sda1`
- Ejecutamos `lilo`: `lilo`

y deberá detectar los arranques que se le indican en `lilo.conf`

2.10.5. Instalación en una máquina que no tiene acceso a cd/dvd

- Instalación por red: del propio servidor de imágenes de Debian hay también la de un cdrom de instalación por red e imágenes de diskettes de 1.44 Mb, permiten hacer la instalación por red contra servidores `http/ftp`
- También se pueden emplear sistemas alternativos (con los mismos cd's o diskettes) como montar una partición/cdrom/dvdrom en otra máquina usando el sistema de ficheros por red (NFS), para ello:
 - el interfaz de red deberá estar configurado
 - desde una consola virtual (se puede lanzar una haciendo `alt+f2`, por ejemplo), se haría algo tal como:
`mount -t nfs 192.168.1.1:/cdrom /mnt`
siendo `192....` la dirección IP de la máquina que tiene el lector de cdrom, `/cdrom` el punto de montaje en esa misma máquina, y `/mnt` el punto de montaje en la máquina en la que se va a hacer la instalación
 - se indica cuando se requiera (fuente para descargar el sistema base, configuración de `apt`) que los paquetes se instalen de la ya montada partición `/mnt`
 - una vez instalado lo mínimo para tener el sistema funcionando, se pueden montar diferentes cd's por `nfs`, y cada vez habrá que hacer un `update` de los paquetes disponibles a continuación

Obviamente, la máquina con cdrom deberá tener instalado soporte para `nfs`, en el kernel y el servidor correspondiente: consultar `nfs-howto` del Linux Documentation Project (<http://www.tldp.org>) para más detalles

- Otra posibilidad es coger el disco duro de la máquina, insertarlo en otra, realizar la instalación normalmente, y modificar después (si es necesario) `/etc/mstab`, donde se indican las particiones que se van a montar al arrancar; y también habrá que reconfigurar X para que funcione con una tarjeta de video diferente si lo es

- Para equipos limitadillos (486-Pentium mmx con poca memoria), es recomendable usar un kernel de la serie 2.4, hay incluso backports de funcionalidades que originalmente se implementaron en 2.6

Capítulo 3

Algunas herramientas y paquetes libres

3.1. Servidores gráficos, gestores de ventanas y escritorios

3.1.1. Servidores gráficos

- XFree86, con el cambio de licencia ya no es libre
- X.org, fork del proyecto anterior, mejoras de arquitectura

3.1.2. Gestores de ventanas

Para sistemas o usuarios que requieren entornos ligeros (aunque más espartanos)

- olwm (recuerdos *vivos* de la Facultad de Informática UDC)
- Window Maker, basado en el gestor original de NexT (a lo que se dedicó Steve Jobs cuando lo echaron de la empresa que había fundado, Apple Computer, además de fundar Pixar)
- After Step, también basado en NexT, desarrollado sobre la plataforma de desarrollo GNU Step
- Enlightenment, IceWM “muy bonitos”
- Muy, muy ligeros: flwm, twm, blackbox, fluxbox (usado en Necromantux)

3.1.3. Escritorios

- Gnome, basado en las bibliotecas de desarrollo GTK, del proyecto The Gimp
- KDE, basado en las bibliotecas de desarrollo Qt, de TrollTech
- xfce, más ligero que los anteriores

Nota: las aplicaciones específicas de un escritorio pueden ser lanzadas desde otro (sólo necesitan tener las bibliotecas de enlace dinámico a mano)

3.2. Red

- Navegadores:
 - Modo texto: lynx, w3m
 - Modo gráfico: mozilla (firefox, navegación con pestañas (tabs), énfasis en cumplir los standards web), konqueror (navegador de kde), nautilus (navegador de gnome), amaya (navegador-editor de páginas web del W3C, muy recomendable para familiarizarse con los standards)
- Clientes de correo:
 - Modo texto: mutt, emacs
 - Modo gráfico: mozilla (thunderbird, lector de RSS integrado), evolution (gnome), kmail (kde)
- Clientes de mensajería instantánea,irc:
 - Modo texto: irssi (irc), bitchx (irc), centericq(múltiples protocolos: irc, icq, jabber, msn, aim...)
 - Modo gráfico: xchat (irc), gaim (gnome, múltiples), kopete (kde, múltiples)
- Descargas masivas:
 - wget descarga vía http (páginas web recursivamente, ficheros, ftp...)
 - amule versión linuxera de emule,edonkey...
 - azureus descargas torrent
- Lectores de news: thunderbird, pan

3.3. Ofimática

- OpenOffice, versión libre de la suite ofimática de Sun
- KOffice, suite ofimática de kde
- latex, un clásico de documentación científica (paquetes tetex-bin, tetex-extra); kile, editor *amigable* de latex
- xpdf (lector de pdf), gv (lector de postscript)

OpenOffice ha establecido y utiliza formatos abiertos de ficheros, además de trabajar nativamente con XML (standard futuro del W3C), por lo que está siendo impulsado en administraciones

3.4. Gráficos

- Dibujo vectorial: inkscape, scribbus
- Retoque fotográfico: The Gimp
- Rendering 3D: blender (usado por los grandes estudios americanos)
- Gestión de scanners: sane
- Gestión de cámaras digitales: gphoto

3.5. Multimedia

- Edición: kine (doméstico), cinelerra (profesional, permite su uso en máquinas distribuidas, edición y rendering *en caliente*)
- Visores: xine (reproduce ficheros de video, dvd, svcd, audio, radio por internet...), totem, xawtv (televisión); para reproducir según que dvd's hay que usar libdvdcss2
- Compresores: ffmpeg, gstreamer
- Edición audio: wavtools, ecawave
- Reproductores de audio: workbone(cd's, modo texto), workman(cd's), mp3blaster (mp3, sencillo, modo texto), amarok (mp3,kde),xmms(mp3,radio...)
- Compresores de audio: oggenc (codificador en el formato abierto ogg), lame (compresor en mp3, restringido por patentes), mpc (reproductor de musepack/mp3+)

3.6. Soportes digitales

- mkisofs (creación de imágenes), cdrecord (grabación de cd's), mp3burn (descompresión+tostado todo en uno), xcdroast (frontend para mkisofs y cdrecord)
- growisofs (paquete dvd+rw-tools, grabación de dvd's, creación de imágenes *on the fly*), k3b (frontend para mkisofs, cdrecord y growisofs)

3.7. Matemáticas

- octave (versión libre de matlab, no tiene tantas bibliotecas como él)
- gmsh mallador elementos finitos
- tochnog programa de cálculo de estructuras (sólo motor, aconsejable pre y postprocesador)
- gnuplot potente programa para generación de varios tipos de gráficas
- grace genera amigablemente gráficos a partir de ficheros de datos

3.8. Ingeniería

- gmt herramientas de generación de mapas a partir de datos empíricos (muy extendido, desarrollado en la Universidad de Hawaii)
- programas de cad:qcad (de la empresa suiza Ribbonsoft,2D), varkon (desarrollado en parte y utilizado por la división aeroespacial de Saab), Open-Cascade (framework de programación de herramientas CAD)
- grass herramientas GIS
- SWAN (Simulating WAVes Nearshore)

3.9. Programación

- Compiladores: proyecto gcc (múltiples lenguajes)
- Depuradores: gdb, ddd(frontend de gdb)
- Sistemas de control de versiones: cvs, subversion (svn), gnuarch, git
- Gestores de compilación: make, automake
- Editores de texto: emacs, xemacs (fork de emacs), vim (versión libre de vi), fte (parecido a los entornos de programación de Borland para ms-dos)

Es importante dedicarle tiempo al principio, al aprendizaje de un buen editor de textos, es tiempo ganado a la hora de trabajar.

Capítulo 4

Dónde descargar software

- El kernel, de <http://kernel.org>
- De cada distribución (o de sus mirrors)
- De la página cada proyecto de software
- De *forjas* de proyectos de software libre:
 - <http://savannah.gnu.org> Forja del proyecto GNU
 - <http://www.sourceforge.net> La madre de todas las forjas, software de desarrollo colectivo del anteriormente conocido como VA Linux Systems, hoy VA Systems
 - <http://www.freshmeat.net> Otra fran forja de proyectos
 - <http://packstormsecurity.org> Alojamiento de paquetes de seguridad
- <http://apt-get.org/> web de repositorios Debian no oficiales
- <http://linuxiso.org/> web de imágenes de distribuciones
- <http://www.rediris.es> la red científica española tiene un interesante servidor ftp

En general, fuera de esos sitios es importante tener en cuenta que un sitio donde no está el código fuente es sospechoso de por sí, también comprobar si las sumas md5 de las imágenes de cd's se cumplen (`md5sum -c fichero_suma.md5`).

Capítulo 5

Documentación y más ayuda

5.1. Documentación

- En las páginas de cada distribución y en los cd's/dvd's
- Dentro del sistema, en las páginas man, obviamente, y en `/usr/share/doc`, donde se aloja la documentación original de cada paquete de software, los cambios entre versiones, la documentación de las modificaciones para la propia distribución y en ocasiones, ejemplos de código o configuración; también se alojan los howtos del tldp si son instalados.
- Proyecto GNU: <http://www.gnu.org>
- The Linux Documentation Project: <http://tldp.org>
- Wide World Web Consortium: <http://www.w3.org>
- OpenDocument: <http://www.oasis-open.org>
- Open Design Alliance: <http://www.opendesign.com/>
- En la página de cada proyecto de software

5.2. Foros de noticias

- Slashdot: <http://slashdot.org>, noticias principalmente sobre software libre, también científicas y derecho en internet
- Barrapunto: <http://barrapunto.com>, lo mismo, pero en español
- Securityfocus: <http://securityfocus.com>, noticias y listas de correo sobre seguridad

- Kriptópolis: <http://kriptopolis.org>, seguridad, criptografía y privacidad, en español
- Schneier on security: <http://www.schneier.com/blog/>, blog del gran experto en criptografía Bruce Schneier
- Electronic Frontier Foundation: <http://www.eff.org/>, fundación para la defensa de la libertad en Internet

5.3. Grupos de usuarios

Desde el principio de GNU/Linux surgieron pequeños y heterogéneos grupos de programadores y usuarios, en casi cualquier ciudad hay uno o varios.

En el caso de Coruña, tenemos al Grupo de Programadores y Usuarios de Linux (GPUL) alojado en la Facultad de Informática: <http://www.gpul.org>. Además del weblog, también mantiene distintas listas de correo donde la gente intercambia problemas y soluciones, además de las típicas *unixwars*.

Sea en GPUL, sea en cualquier otro foro, es importante aportar el mayor número de datos posibles en una consulta: máquina, distribución, ficheros de configuración del programa problemático, volcado de errores...

Por mencionar otro lugar interesante, no por cercanía sino por su actividad, los baleares de Bulma: <http://bulma.org>, destacan por la cantidad de documentación que generan para resolver problemas muy concretos.

Capítulo 6

Taller: Instalación de Mandriva 10.2

Enter...enter...selección de nivel de seguridad...enter...enter... selección de lotes de paquetes...enter...enter...¡ya está!

Apéndice A

Configuración wireless para udcwifi

Ahora mismo es recomendable bajarse la versión de wpa supplicant de la página del propio proyecto (<http://hostap.epitest.fi/wpa-supPLICANT/>), y parchearlo con el parche que se proporciona en la página de udcwifi (<http://www.udc.es/udcwifi>), guardando el fichero `eap_success_workaround.patch` en el directorio del supplicant descomprimido, y se ejecuta:

```
patch < eap_success_workaround.patch
```

A continuación se edita el fichero de ejemplo `defconf`, y se guarda como `.config`:

```
# Example wpa_supplicant build time configuration
#
# This file lists the configuration options that are used when building the
# hostapd binary. All lines starting with # are ignored. Configuration option
# lines must be commented out complete, if they are not to be included, i.e.,
# just setting VARIABLE=n is not disabling that variable.
#
# This file is included in Makefile, so variables like CFLAGS and LIBS can also
# be modified from here. In most cass, these lines should use += in order not
# to override previous values of the variables.

# Uncomment following two lines and fix the paths if you have installed openssl
# in non-default location
#CFLAGS += -I/usr/local/openssl/include
#LIBS += -L/usr/local/openssl/lib

# Example configuration for various cross-compilation platforms

#### sveasoft (e.g., for Linksys WRT54G) #####
#CC=mipsel-uclibc-gcc
```

```

#CC=/opt/brcm/hndtools-mipsel-uclibc/bin/mipsel-uclibc-gcc
#CFLAGS += -Os
#CPPFLAGS += -I../src/include -I../src/router/openssl/include
#LIBS += -L/opt/brcm/hndtools-mipsel-uclibc-0.9.19/lib -lssl
#####

#### openwrt (e.g., for Linksys WRT54G) #####
#CC=mipsel-uclibc-gcc
#CC=/opt/brcm/hndtools-mipsel-uclibc/bin/mipsel-uclibc-gcc
#CFLAGS += -Os
#CPPFLAGS=-I../src/include -I../openssl-0.9.7d/include \
# -I../WRT54GS/release/src/include
#LIBS = -lssl
#####

# Driver interface for Host AP driver
CONFIG_DRIVER_HOSTAP=y

# Driver interface for Agere driver
#CONFIG_DRIVER_HERMES=y
# Change include directories to match with the local setup
#CFLAGS += -I.././hcf -I.././include -I.././include/hcf
#CFLAGS += -I.././include/wireless

# Driver interface for madwifi driver
CONFIG_DRIVER_MADWIFI=y
# Change include directories to match with the local setup
CFLAGS += -I../madwifi

# Driver interface for Prism54 driver
#CONFIG_DRIVER_PRISM54=y

# Driver interface for ndiswrapper
#CONFIG_DRIVER_NDISWRAPPER=y

# Driver interface for Atmel driver
#CONFIG_DRIVER_ATMEL=y

# Driver interface for Broadcom driver
#CONFIG_DRIVER_BROADCOM=y
# Example path for wlioctl.h; change to match your configuration
#CFLAGS += -I/opt/WRT54GS/release/src/include

# Driver interface for Intel ipw2100/2200 driver
#CONFIG_DRIVER_IPW=y

```

```
# Driver interface for generic Linux wireless extensions
CONFIG_DRIVER_WEXT=y

# Driver interface for FreeBSD net80211 layer (e.g., Atheros driver)
#CONFIG_DRIVER_BSD=y
#CFLAGS += -I/usr/local/include
#LIBS += -L/usr/local/lib

# Driver interface for Windows NDIS
#CONFIG_DRIVER_NDIS=y
#CFLAGS += -I/usr/include/w32api/ddk
#LIBS += -L/usr/local/lib
# For native build using mingw
#CONFIG_NATIVE_WINDOWS=y
# Additional directories for cross-compilation on Linux host for mingw target
#CFLAGS += -I/opt/mingw/mingw32/include/ddk
#LIBS += -L/opt/mingw/mingw32/lib
#CC=mingw32-gcc

# Driver interface for development testing
#CONFIG_DRIVER_TEST=y

# Enable IEEE 802.1X Supplicant (automatically included if any EAP method is
# included)
CONFIG_IEEE8021X_EAPOL=y

# EAP-MD5 (automatically included if EAP-TTLS is enabled)
CONFIG_EAP_MD5=y

# EAP-MSCHAPv2 (automatically included if EAP-PEAP is enabled)
CONFIG_EAP_MSCHAPV2=y

# EAP-TLS
CONFIG_EAP_TLS=y

# EAP-PEAP
CONFIG_EAP_PEAP=y

# EAP-TTLS
CONFIG_EAP_TTLS=y

# EAP-GTC
CONFIG_EAP_GTC=y

# EAP-OTP
```

```

CONFIG_EAP_OTP=y

# EAP-SIM (enable CONFIG_PCSC, if EAP-SIM is used)
#CONFIG_EAP_SIM=y

# EAP-PSK (experimental; this is not needed for WPA-PSK)
#CONFIG_EAP_PSK=y

# LEAP
CONFIG_EAP_LEAP=y

# EAP-AKA (enable CONFIG_PCSC, if EAP-AKA is used)
#CONFIG_EAP_AKA=y

# PKCS#12 (PFX) support (used to read private key and certificate file from
# a file that usually has extension .p12 or .pfx)
CONFIG_PKCS12=y

# PC/SC interface for smartcards (USIM, GSM SIM)
# Enable this if EAP-SIM or EAP-AKA is included
#CONFIG_PCSC=y

# Development testing
#CONFIG_EAPOL_TEST=y

# Replace native Linux implementation of packet sockets with libdnet/libpcap.
# This will be automatically set for non-Linux OS.
#CONFIG_DNET_PCAP=y

# Include control interface for external programs, e.g, wpa_cli
CONFIG_CTRL_IFACE=y

# Include interface for using external supplicant (Xsupplicant) for EAP
# authentication
#CONFIG_XSUPPLICANT_IFACE=y

# Include support for GNU Readline and History Libraries in wpa_cli.
# When building a wpa_cli binary for distribution, please note that these
# libraries are licensed under GPL and as such, BSD license may not apply for
# the resulting binary.
#CONFIG_READLINE=y

```

Esta es mi configuración para mi caso, que tengo un chip Atheros en la minipci de mi portátil, habría que hacer cambios para cada caso de chip (ipw, prism, ndiswrapper-o como convertir un driver para windows en driver de linux...).

Y a continuación se construyen los ejecutables `wpa_supplicant` y `wpa_cli`, ejecutando:

```
make
```

copiando a continuación los dos binarios en `/usr/local/bin`

La configuración, en el fichero `/etc/wpa_supplicant.conf`:

```
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
eapol_version=1
ap_scan=1

network={
ssid='udcwifi'
key_mgmt=WPA-EAP
pairwise=TKIP
group=TKIP
eap=TTLS
identity='camfts00@udc.es'
password='contraseña'
priority=2
}
```

Para entablar la conexión, ejecutamos:

```
wpa_supplicant -iwlan0 -c/etc/wpa_supplicant.conf -d
```

si queremos ver el diagnóstico del proceso, o:

```
wpa_supplicant -iwlan0 -c/etc/wpa_supplicant.conf -B
```

si lo que queremos es tenerlo ejecutando en *background*. Cualquiera de las dos opciones hará una negociación con el punto de acceso adecuado, y una vez establecida la conexión, se lanza un cliente de dhcp:

```
dhclient ath0
```

ya que en mi caso el driver del Atheros genera un interfaz llamado `ath0`.

Y ya se podrían hacer conexiones, vía `http` o `ftp` (o si uno es más atrevido, se puede hacer un tunel `ssh` sobre `http` para poder usar el protocolo que nos de la gana). En cada consola, no se pueden olvidar las variables de entorno para poder conectar a través del proxy:

```
export http_proxy="http://10.3.1.2:3128"
```

```
export ftp_proxy="http://10.3.1.2:3128"
```

En otras herramientas, como la suite mozilla o de escritorio, el proxy se indica en los menús de configuración de ellas mismas o del escritorio correspondiente.