

FENT LINUX

Año 2005 Nº 1



magazine

Paquetes .deb

**Notas grub
y lilo**

**Notas de
instalación
de Slax**

**Entrevista
Molinux**

Recetario APT



**Howto
Radio Online**

FENT LINUX



**El resumen
de un año**

**1er Aniversario
FENTLINUX**



UN AÑO EN UNAS PALABRAS

Allá por el mes de marzo tres amigos deciden iniciar un proyecto dedicado al software libre, dan inicio las probaturas, barajamos opciones, y al final nos decantamos por el *CMS del Post Nuke* convirtiéndonos en un portal linux de actualidad y documentación.

Gracias a Antonio de la Flor, webmaster entre otros espacios de *adelaflor.com*, que nos cede espacio en su servidor, iniciamos andadura durante la primera semana de abril, siendo el 4 del mismo mes el día de la publicación del primer artículo/noticia.

El proyecto fue iniciado por **gringo, halo y Paulet**, con la estimable colaboración de *thyyzar*, nuestro diseñador gráfico personal y creador de los *fentux's*, esos fabulosos gif únicos en la red.

Un mes y medio después abrimos un *foro* con el sistema de *phpBB*, más concretamente el 17 de mayo, con apenas mes y medio en la red, y entraron a formar parte del equipo y por ende de la comunidad administrativa **davidfm, kalli y destroyer**.

Desde entonces el foro ha sufrido diversas modificaciones con el fin de adaptarlo a nuestras y vuestras necesidades. Actualmente cuenta con 20 secciones de diversa índole, desde generales a específicas de cada distribución, pasando por la propias y necesarias que recogen todos los hechos y proyectos acontecidos en **FentLinux**.

Agosto fue otro mes importante en la corta intensa trayectoria de **FentLinux**, en él **Zx80 y vl@d** se sumaron al proyecto ya pasaron a moderar y administrar diversas secciones del foro.

Y a finales, el 29 de agosto, se produjo la adquisición de un dominio específico para **FentLinux**, siendo entonces nuestra nueva dirección: <http://www.fentlinux.com>.

A finales de noviembre dotamos al espacio de una nueva funcionalidad con un *wiki* basado en *mediawiki*, *wiki* que bautizamos como el **Centro de Documentación FentLinux** y que concretamente abrió sus puertas un 24 de noviembre, con más de 120 documentos y albergando el proyecto de *faq's* aún en construcción.

El **wiki**, justo antes de la navidad paso por un periodo de crisis, 21 de diciembre, crisis subsanada el 14 de febrero de 2005.

Actualmente da cabida a una gran cantidad de documentos originales y confeccionados por los miembros de la comunidad y alberga todos nuestros proyectos en fase desarrollo: *Hardware Por Categorías; Faq's; Recopilación Live's; Parches kernel; Traducciones; Extensiones Firefox...*

Febrero también fue el mes de **grotfang**, que intensifico su colaboración pasando a moderar diversas secciones del foro y aumentando de esta forma el número de responsables que velan por el buen funcionamiento de la comunidad.

Reseñar el mes de enero como el mes solidario, ya que **FentLinux**, creyó oportuno no sólo preocuparse por lo que acontece en el mundo linux si no sumarse a la solidaridad que debería reinar entre todos,



naciendo de esa forma el *Rincón Solidario*.

En el mismo mes se votaron, eligieron candidatos y salieron premiados firefox, comunidades andaluzas y extremeñas y ubuntu, con la *1ª edición de los premios Fentlinux*.

Acabamos el año con tres subespacios, actualizados todos los días, con más de 1000 noticias/artículos, más de 150 documentos y 5 proyectos en marcha, tantos usuarios y muchas visitas a nuestras espaldas.

Durante todo este tiempo **FentLinux** se ha mostrado como una comunidad activa, preocupados por la actualidad y por ofrecer siempre lo mejor de sí mismos, con muchos proyectos e ilusiones, con ideas constantes de mejora y un enorme deseo de que todos un día seamos un poco más libres y un gran respeto por todo y por todos.

Con el deseo de que este primer año sea el primero de una larga serie de años sean bienvenidos al mundo **FentLinux**, seguro que muy pronto os depararemos nuevas sorpresas, algo nuevo, o algo que nos abandonará, mientras todo esto sucede, larga vida a **FentLinux**.

Finalizo reseñando unas direcciones para el recuerdo de los nostálgicos entre los que me encuentro:

La primera dirección web de **FentLinux**

<http://adelaflor.com/fentlinux>

Y sus primeros foros:

www.adelaflor.com/fentlinux/foros

<http://adelaflor.com/fentlinux/foros>





ENTREVISTA A MOLINUX

En los últimos tiempos estamos viendo todo un auge de distribuciones de Linux en España, el camino iniciado por Linex en Extremadura en el campo de las Administraciones Públicas esta siendo seguido por otras Comunidades Autónomas que también han preparado su proyecto de distribución.

Una de ellas es Molinux, la versión de Linux manchega ha irrumpido hace poco en escena y se presenta como un proyecto fuerte y activo para llevar a todos los castellano-manchegos un sistema operativo fácil de utilizar con todo aquello que el usuario informático necesita hoy en día.

Fentlinux se puso en contacto con D. Rafael Ariza Fernández, Director General para la Sociedad de la Información y las Telecomunicaciones de la Junta de Comunidades de Castilla-La Mancha y le realizó la siguiente entrevista que ahora reproducimos aquí.

FENTLINUX: Hay un pequeño temor en la comunidad linuxera de que todas estas nuevas distribuciones territoriales no sean mas que una moda iniciada por Linex, ¿es molinux un proyecto serio de la Junta de Comunidades o por el contrario es una iniciativa que suma a la moda de tener una distribución propia?

MOLINUX: Por supuesto que es una iniciativa seria. De hecho es un paso en una estrategia para el fomento del uso del software libre en Castilla-La Mancha.

FENTLINUX: ¿Cuándo surge la idea? ¿Y cuándo tardan en llevarla a cabo?

MOLINUX: Aunque la idea surge hace un par de años, es con el nacimiento de la Consejería de Industria y Tecnología cuando se apuesta por Linux como una vía de desarrollo de la Sociedad de la Información y el Conocimiento y se elabora la estrategia a seguir. El proceso de creación y lanzamiento de Molinux es de aproximadamente 4 meses.

FENTLINUX: Una de las cosas que mas ha llamado la atención es el nombre que se le ha puesto a la distribución, ¿como se decidió?

MOLINUX: El nombre surgió en una tormenta de ideas entre el equipo de seguimiento del proyecto. Cómo suele pasar en estas cuestiones empezó siendo un nombre interno del proyecto y acabó como nombre definitivo.

Pensamos que el nombre identifica muy rápidamente la región de origen con el producto, aunando aspectos clásicos en la iconografía de Castilla-La Mancha con los aspectos de tecnología y evolución



que presenta un proyecto Linux.

FENTLINUX: ¿Por qué debian?

MOLINUX: Primero por coherencia y homogeneidad con las distribuciones hermanas de Extremadura y Andalucía. Si queríamos aprovechar lo ya andado por estas distribuciones y lograr una compatibilidad a todos los niveles lo más adecuado es utilizar Debian.

En segundo lugar los aspectos tecnológicos también son importantes. El tipo de licencia Debian nos permite asegurar la permanencia en el futuro de la distro en los mismos términos de libertad de código.

FENTLINUX: Actualmente, ¿cuántas personas o desarrolladores hay dedicadas al proyecto?

MOLINUX: Depende de las fases del proyecto. En principio hubo 6 desarrolladores directamente implicados en una u otra forma en el proyecto. En la actualidad hay 4 personas dedicadas a Molinux. Además hay una mesa técnica de seguimiento del proyecto compuesta por 8 técnicos.



FENTLINUX: ¿El proceso de implantación en colegios y bibliotecas va a ser lento? ¿Para cuando?

MOLINUX: La implantación en la Red de Centros Internet de la Junta de Comunidades de Castilla-La Mancha ya se está produciendo. Se están instalando todos los centros con arranque dual para que sea el usuario quien elija que sistema operativo y herramientas quiere correr en su sesión.

En cuanto a los centros educativos será en el próximo curso escolar cuando se vaya desplegando Molinux.

FENTLINUX: ¿Van a recibir por parte de la administración manchega formación los profesores, alumnos y profesionales informáticos en torno al manejo de Molinux?

MOLINUX: Se está preparando un ambicioso plan de formación para toda la comunidad castellano-manchega que abarca desde cursos de introducción a molinux, cursos avanzados y cursos de formación de formadores.

FENTLINUX: Suponemos que una de las cosas que mas les preocupará es en nivel de aceptación de la distro tanto en la comunidad manchega como fuera de ella. ¿Como ha sido?



MOLINUX: El nivel de aceptación en Castilla-La Mancha ha sido alto. Se han distribuido directamente cerca 10.000 CD de la versión 1.0 y se han producido 7.000 descargas de la web. Estos son números muy altos si tenemos en cuenta la penetración de las tecnologías en la Región.

También ha sido buena la reacción fuera de la comunidad autónoma, baste indicar el interés de una revista del sector que en su número del mes de abril repartirá 60.000 ejemplares de la versión de Molinux 1.0.1 junto con su tirada.

FENTLINUX: ¿Han pensado en la posibilidad de lanzar un live-CD, tan de moda en la actualidad?

MOLINUX: Aunque se barajó en un principio la idea, se descartó para inicialmente centrarnos en una versión de instalación completa.

No descartamos tener una versión en live-cd a medio plazo.

FENTLINUX: ¿Contentos con el número de descargas de Molinux?

MOLINUX: Sí, si añadimos al las 7.000 descargas en 3 meses la distribución de los casi 10.000CD que se ha hecho directamente de la versión 1.0 tenemos que mostrarnos contentos de la acogida recibida.

FENTLINUX: ¿Cuántos nuevos usuarios linux gracias a molinux?

MOLINUX: Es una cuestión muy difícil de valorar. Por las llamadas que recibimos en el centro de soporte podemos calcular que un 10% de las personas que consultan no son usuarios linux.



FENTLINUX: Y a partir de ahora, ¿cuales son los planes de futuro de Molinux?

MOLINUX: Para este año se prevén sacar dos revisiones nuevas de molinux, una en junio y otra en octubre. La principal novedad es que estas distribuciones se van a basar en Ubuntu con quien estamos a punto de llegar a un acuerdo de distribución basada en su plan de versiones.

FENTLINUX: Ya que ha salido Ubuntu en la conversación... ¿Cuales son aspectos fundamentales del acuerdo alcanzado con ellos?

MOLINUX: Nos encontramos en fase aún de redacción del protocolo. El aspecto básico es asegurarnos la continuidad del acuerdo de licencia para que el proyecto continúe siendo libre 100% y no se separen en el futuro de Debian.



FENTLINUX: ¿Y como van las relaciones con las demas comunidades autónomas con distribución propia?

MOLINUX: Existe un grupo de trabajo entre administraciones públicas con distro de Linux que se encarga de trabajar en poner de común acuerdo a todas las administraciones. Asimismo se acaba de dar luz verde a un proyecto de centro de coordinación que liderará Extremadura y que servirá para aunar aún más los esfuerzos que en el campo del software libre realizamos todas las administraciones.

FENTLINUX: Por cierto, ¿os habeis encontrado alguna traba por parte de organismos, instituciones, individuos a la hora de desarrollar la distro?

MOLINUX: No nos hemos encontrado grandes trabas. Sí es cierto que cada grupo de interés se defiende aportando al debate sus argumentos pero dado que nuestra posición no es beligerante contra nadie sino que pretendemos escoger la mejor solución para cada proyecto, estas aportaciones no se han traducido en presiones sino más bien en ideas que nos están ayudando a tener criterios de selección para saber en qué caso es mejor cada solución.



FENTLINUX: A parte de crear una distribución propia ¿qué más cambios o medios están previstos?

MOLINUX: En reciente comparecencia en Cortes de Castilla-La Mancha la Consejería de Industria y Tecnología se ha comprometido ha elaborar un plan que fomente el uso del software libre en la Región a todos los niveles: en el ámbito educativo, en el ámbito sanitario, en al ámbito administrativo y también en el ámbito privado y empresarial.

Es una labor de divulgación que nos toca seguir haciendo en la Región, que es nuestro ámbito competencial, y que ayude a mantener la independendencia tecnológica y asegure la libertad de elección.

FENTLINUX: Para finalizar, ¿que esperáis aportar a la comunidad del software libre? ¿y que esperáis de la comunidad libre?

MOLINUX: Nuestra principal aportación esperamos que sea impulso del software libre y de las empresas TIC de Castilla-La Mancha. Queremos fomentar nicho de negocio para estas pequeñas empresas TIC que existen en nuestra Región para las cuales el paradigma del software libre es sin duda una oportunidad y un reto.



De la comunidad de software libre, sobre todo regional, esperamos su movilización y apoyo, no sólo con su esfuerzo, que ya lo hacen sino con sus ideas que nos van a ayudar a hacer más valioso el proyecto.

El equipo Fentlinux quiere agradecer enormemente su colaboración tanto al señor D. Rafael Ariza Fernández como a los webmasters y servicio de información de Molinux para la realización de esta entrevista.

Gracias por su tiempo e interés.



NOTAS DE INSTALACIÓN SLAX.

Por: Grotfang

Arrancamos el ordenador con el live metido, entrando así en Slax. Cuando acabe de cargar, nos aparecerá el prompt pidiéndonos un login y un password. Normalmente lo tendréis en la parte superior derecha escrito, para entrar como root y para entrar como usuario normal. Entramos como root:

```
$ root  
$ ****
```

Ahora, una vez logueados como root, entramos al entorno gráfico. Para KDE:

```
$ gui
```

Para fluxbox:

```
$ guifast
```

Bueno, yo he entrado en KDE. Una vez cargado el entorno gráfico, vamos a poner el teclado en español. Para eso accedemos al centro de control (es un icono formado por dos herramientas, situado en la parte inferior de la pantalla), pinchamos en Regional&Accessibility, después en Keyboard Layout, elegimos Spanish de la lista de la izquierda y pinchamos en "Add", luego pulsamos en "Apply" y salimos del centro de control. Ahora pinchamos con el botón derecho del ratón sobre la bandera de EEUU que aparece en la esquina inferior derecha y seleccionamos la banderita española. Ya tenemos el teclado en español, con lo que ya podemos pasar sin problemas a la instalación (también se puede instalar con el teclado en ingles, pero nos será más cómodo en español).

INSTALACIÓN

Abrimos la consola de comandos y escribimos cfdisk:

```
$ cfdisk
```

Pulsamos intro. Veremos una pantalla en la que se nos mostrará las particiones actuales de nuestro disco duro y su sistema de archivos. Yo tengo 3 particiones, una para swap, otra de arranque y la raíz.



Como voy a explicarlo desde el principio, elimino todas las particiones y me quedo con los 10 Gb de mi disco duro totalmente libres, luego cada uno de se reparta su disco duro como quiera.

Vamos seleccionando cada una de las particiones con las flechas de arriba y abajo y vamos pulsando, con las flechas de izquierda y derecha, en delete con cada una de ellas. Una vez hecho veremos como todo lo que tenemos de espacio en el disco duro se nos queda como "free space". Bueno, damos a "write" para guardar los cambios. Ahora vamos a seleccionar "new" para crear una partición nueva, que la seleccionaremos como primaria, y le asignaremos 16.00 Mb. Esta partición será el /boot. Para asignar el tamaño, cuando nos aparezca los Mb libres en nuestro disco duro disponibles, escribimos el tamaño que queramos, separando los enteros con un punto y pulsando intro. Una vez creado el /boot, pulsamos intro teniendo la casillita posicionada sobre "bootable", hasta que veamos que en la columna flags, en esa partición, nos aparece la palabra "boot". Bueno, ahora vamos a crear la partición swap. Damos a "new" y seleccionamos una partición lógica, le asignamos de tamaño, como mínimo, el doble de lo que tengamos de RAM. Yo le voy a asignar 512.00Mb, y tengo 80 de RAM. Una vez creada, pulsamos en "type", nos aparecerá una nueva pantallita que nos dirá que pulsemos una tecla para continuar, lo hacemos y nos pedirá que metamos el número correspondiente al formato que queramos darle a la partición que hemos creado para la RAM (los números que aparecen están en hexadecimal). Escribimos 82 (si es que no aparece como predeterminado), que es el número que corresponde a Linux Swap y pulsamos intro. Ya tenemos creada la partición de swap. Volvemos a nuestro espacio que todavía nos queda libre y volvemos a seleccionar "new", también volvemos a escoger una partición lógica, y le dejamos asignado el tamaño que nos sobre de disco duro. Esta será la partición raíz (/root). Guardamos todos los cambios pulsando sobre "write", como os he dicho antes. Lo suyo es que por cada cambio que hagáis, lo guardéis (asegurándoos primero que lo habéis hecho bien, claro). Os habéis tenido que haber quedado con 3 particiones, hda5, hda1 y hda6 (puede variar según el caso), que son /root, /boot y swap respectivamente. Si creéis que está todo correcto, damos a "quit" para salir del cfdisk. Os dejo una captura de como queda mi disco duro después de haber hecho los ajustes correspondientes.

```
Shell - Konsole
cfdisk 2.12g

Disk Drive: /dev/hda
Size: 10205282304 bytes, 10.2 GB
Heads: 255 Sectors per Track: 63 Cylinders: 1240

-----
Name      Flags      Part Type FS Type      [Label]      Size (MB)
-----
hda5      Logical   Linux ext3   9672.9*
hda6      Logical   Linux swap   509.97
hda1      Boot      Primary Linux ext3   16.46

[Bootable] [ Delete ] [ Help ] [Maximize] [ Print ]
[ Quit ] [ Type ] [ Units ] [ Write ]

Toggle bootable flag of the current partition
```



Ahora lo que tenemos que hacer es dar formato a las dos particiones que nos quedan sin dárselo, /boot y /root. Para ello usaremos el comando `mke2fs -j` para cada partición. Así:

```
$ mke2fs -j /dev/hda1  
$ mke2fs -j /dev/hda5
```

Ya tenemos las dos particiones formateadas en ext3. Ahora vamos a crear el lugar donde se va a realizar la instalación:

```
$ mkdir /mnt/partslax
```

Ahora la montaremos junto con la partición raíz:

```
$ mount /dev/hda5 /mnt/partslax
```

Ahora vamos a crear otra carpetita dentro de esa:

```
$ mkdir /mnt/partslax/boot
```

Y la montaremos junto con la partición de arranque:

```
$ mount /dev/hda1 /mnt/partslax/boot
```



Bueno, pues ya hemos preparado el terreno, solo queda el comando para instalar, pero no tan rápido, primero tendremos que reiniciar, si no lo hacemos no seremos capaces de instalar.

Una vez reiniciado el ordenador, y otra vez dentro de slax y con el teclado en español (por comodidad, ya sabéis), escribimos lo siguiente en la consola de comandos:

```
$ slax-install
```

Pulsamos intro y.... voilá, un pequeño asistente que nos pide que seleccionemos una partición. A mi me han aparecido dos, y he elegido "disc0part5", le he dado a intro y después de unos minutos me ha aparecido un mensajito muy chulo que decía:

Installation succesfull

Lo que viene a significar que la instalación ha sido un éxito, así que he intentado encender el ordenador con el cd del live quitado y... ya tengo slax funcionando perfectamente en mi ordenador. El funcionamiento es el mismo que el del live cd, pero con la ventaja que supone tenerlo instalado en el



ordenador. Ahora es cuestión de ir probando y ver hasta donde puede llegar este live instalado en un disco duro.

CONFIGURACIÓN DEL SONIDO

Slax tiene muy buena detección de hardware, pero en los ordenadores que he trabajado con el live, en ninguno me ha reconocido la tarjeta de sonido al principio, he tenido que darle un empujoncito y luego sí, sin problema. Si a vosotros tampoco os ha reconocido la tarjeta de sonido, os explico como podemos guiarle para que nos la reconozca:

Abrimos la consola de comandos y escribimos:

```
$ alsacnf
```

Pulsamos intro y entraremos en una especie de "asistente" para la configuración. En la primera pantallita que nos aparecerá le damos a OK porque no es nada relevante. Esperamos a que realice la búsqueda del hardware... Cuando haya acabado, nos encontraremos ante esta ventana:



Bien, vemos que hay dos opciones, la primera es el resultado de la búsqueda automática de hardware realizada por el sistema, a no ser que estemos completamente seguros de que no es la selección adecuada, dejamos esa opción seleccionada y pulsamos intro. Nos aparecerá un mensaje advirtiéndonos de que va a haber cambios en el fichero /etc/modules.conf, damos a YES y a esperar que termine la instalación. Ya deberíamos poder escuchar por nuestros altavoces.

Si por el contrario, la búsqueda automática no ha dado con el resultado adecuado, podemos ver la lista de drivers preparados para instalar en el sistema, simplemente tenemos que escoger la segunda opción



que nos aparece y dar a OK. Nos encontramos ante una pantalla como esta:



Seleccionamos la opción adecuada para nuestra tarjeta de sonido y damos a OK. Esperamos a que acabe la instalación y ya está. Si queremos comprobar si funciona bien, podemos usar un programilla que viene en esta distribución, el mpg321:

```
$ mpg321 /ruta/atu/archivo.mp3
```

Si lo escuchas, felicidades, lo has conseguido.

Si por lo que sea, tus drivers no se encuentran en esta distribución, tendrás que bajártelos e instalártelos manualmente.

CONFIGURAR EL APAGADO DEL SISTEMA

Seguramente ya os habréis dado cuenta de que si, dentro de un sistema gráfico, intentamos apagar el ordenador dando a "logout...", lo que nos hará será sacarnos del entorno gráfico y dejarnos en el prompt de arranque. Bien, pues esto hay manera de solucionarlo para que, dando a "logout...", nos apague el ordenador.

Nos vamos al centro de control (ya sabéis, el icono situado en la parte inferior que está compuesto por dos herramientas), pinchamos en la sección "KDE Components", ahora vamos a "Session Manager".



Nos fijamos en la sección "Default Shutdown Option" y seleccionamos la opción "Turn off computer", aplicamos y ya está, soluciona el apagado del ordenador.





DPKG-DEB

Por: Paulet y vl@d

En la evolución de GNU/Linux ha habido un camino un tanto tortuoso a veces en la instalación de programas. Todavía hoy podemos bajarnos las fuentes de un programa y compilarlas para nuestro sistema (la mejor opción sin duda), pero las más modernas distribuciones nos ofrecen su propio sistema de paquetes precompilados y como no, Debian no iba a ser menos.

La utilidad debianita que ahora nos ocupa es dpkg. Dpkg es el instalador de paquetes Debian por defecto, aunque como ya sabéis, hay otra utilidad llamada “apt” para la instalación de software vía internet.

Un paquete Debian tiene una nomenclatura más o menos estándar, he aquí un ejemplo:

```
fentlinux_0.9.0rc2-1-7_i386.deb
```

Donde podemos ver el nombre, versión y revisión del programa, así como la arquitectura para la que esta compilado (en este caso la i386). Y por supuesto terminado en “.deb”.

Un vistazo al comando “dpkg -h” nos puede dar una idea de la potencia del programa:

```
user@ubuntu:~ $ dpkg -h
```

Usage:

```
dpkg -i/--install <.deb file name> ... | -R/--recursive <dir> ...  
dpkg --unpack <.deb file name> ... | -R/--recursive <dir> ...  
dpkg -A/--record-avail <.deb file name> ... | -R/--recursive <dir> ...  
dpkg --configure <package name> ... | -a/--pending  
dpkg -r/--remove | -P/--purge <package name> ... | -a/--pending  
dpkg --get-selections [<pattern> ...] get list of selections to stdout  
dpkg --set-selections set package selections from stdin  
dpkg --update-avail <Packages-file> replace available packages info  
dpkg --merge-avail <Packages-file> merge with info from file  
dpkg --clear-avail erase existing available info  
dpkg --forget-old-unavail forget uninstalled unavailable pkgs  
dpkg -s/--status <package-name> ... display package status details  
dpkg -p/--print-avail <package-name> ... display available version details
```



dpkg -L/--listfiles <package-name> ... list files `owned' by package(s)

dpkg -l/--list [<pattern> ...] list packages concisely
dpkg -S/--search <pattern> ... find package(s) owning file(s)

dpkg -C/--audit check for broken package(s)
dpkg --print-architecture print target architecture (uses GCC)
dpkg --print-gnu-build-architecture print GNU version of target arch
dpkg --print-installation-architecture print host architecture (for inst'n)
***dpkg --compare-versions <a> <rel> *** compare version numbers - see below
dpkg --help | --version show this help / version number
dpkg --force-help | -Dh/--debug=help help on forcing resp. debugging

dpkg --licence print copyright licensing terms

Use ***dpkg -b/--build|-c/--contents|-e/--control|-I/--info|-f/--field|-x/--extract|-X/--vextract|--fsys-tarfile*** on archives (type ***dpkg-deb --help.***)

For internal use: ***dpkg --assert-support-predepends | --predep-package | --assert-working-epoch | --assert-long-filenames | --assert-multi-conrep***



Options:

- admindir=<directory>*** Use *<directory>* instead of */var/lib/dpkg*
- root=<directory>*** Install on alternative system rooted elsewhere
- instdir=<directory>*** Change inst'n root without changing admin dir
- O/--selected-only*** Skip packages not selected for install/upgrade
- E/--skip-same-version*** Skip packages whose same version is installed
- G/--refuse-downgrade*** Skip packages with earlier version than installed
- B/--auto-deconfigure*** Install even if it would break some other package
- no-debsig*** Do no try to verify

package signatures



--no-act/--dry-run/--simulate

Just say what we would do - don't do it

-D/--debug=<octal> *Enable debugging - see -Dhelp or --debug=help*

--status-fd <n> *Send status change updates to file descriptor <n>*

--ignore-depends=<package>,... *Ignore dependencies involving <package>*

--force-... *Override problems - see --force-help*

--no-force-.../--refuse-... *Stop when problems encountered*

--abort-after <n> *Abort after encountering <n> errors*

Comparison operators for --compare-versions are:

lt le eq ne ge gt (treat empty version as earlier than any version);

lt-nl le-nl ge-nl gt-nl (treat empty version as later than any version);

<<< <= = >= >>> (only for compatibility with control file syntax).

Use 'dselect' or 'aptitude' for user-friendly package management.

Vamos a centrarnos en los comandos más habituales que un usuario de un sistema Debian o basado en él tendrá que ejecutar. para instalar un paquete Debian:

dpkg -i nombre_paquete.deb

Si hubiera algún mensaje de error estos se nos mostrarían en pantalla.

Para forzar la instalación y no ver ningún mensaje de error:

dpkg -force-all nombre_paquete

Sin el numero de versión ni nada. No es muy recomendable esta opción, el sistema dpkg nos proporciona una instalación de paquetes ordenada y limpia, y actuaciones como esta pueden llevarnos de cabeza más tarde.

Para desinstalar, pero sin borrar los archivos de configuración:

dpkg -r nombre_paquete

Con la opción *-purge* limpiamos todos los ficheros de configuración:



dpkg -P nombre_paquete

Para mostrar un listado de todos los paquetes instalados en el sistema, junto con la versión y una breve descripción:

dpkg -l

Para obtener la información de un paquete en concreto:

dpkg -l nombre_paquete

Que ficheros contiene un paquete determinado:

dpkg -L nombre_paquete

Para averiguar a que pertenece un fichero en tu ordenador:

dpkg -S fichero

La tabla de opciones, sacada directamente de 'man dpkg', que os puede resultar muy útil:

dpkg -b | --build directorio [fichero] Construye un paquete Debian GNU/Linux.

dpkg -c | --contents fichero Muestra el contenidos de un paquete Debian GNU/Linux.

dpkg -e | --control fichero [directorio] Extrae la información de control de un paquete.

dpkg -x | --extract fichero directorio Extrae los ficheros contenidos en el paquete.

dpkg -f | --field fichero [campo-control] ... Muestra el/los campo(s) de control de un paquete.

dpkg --fsys-tarfile fichero Muestra el fichero tar contenido en el paquete Debian.

dpkg -I | --info fichero [fichero-control] Muestra información sobre el paquete.

dpkg -X | --vextract fichero directorio Extrae y muestra los nombres de ficheros contenidos en un paquete.



dpkg -l | --list patrón-nombre-paquete ...Lista los paquetes cuyo nombre encaja en el patrón dado.

dpkg -s | --status nombre-paquete ...Informa del estado del paquete especificado.

dpkg -L | --listfiles paquete ...Lista los ficheros instalados en el sistema, que pertenecen a paquete.

dpkg -S | --search patrón-búsqueda-ficheros ...Busca un fichero en los paquetes instalados.

dpkg -p | --print-avail paquete ...Imprime información sobre el paquete, sacada de /var/lib/dpkg_/available.



RECETARIO APT

Por: Paulet

apt-get update

Actualiza el listado de paquetes disponibles.

apt-get check

Comprueba que todo ha ido bien tras la utilización de apt-get update.

```
root@fentlinux:~# apt-get update
Necesito descargar 4684B de archivos.
Se utilizarán 17,4kB de espacio de disco adicional después de desempaquetar.
Des:1 ftp://ftp.us.debian.org testing/main xfce4 4.0.5-1 [4684B]
Descargados 4684B en 3s (1243B/s)
Seleccionando el paquete xfce4 previamente no seleccionado.
(Leyendo la base de datos ...)
$2209 ficheros y directorios instalados actualmente.
Desempaquetando xfce4 (de ../archives/xfce4_4.0.5-1_all.deb) ...
Configurando xfce4 (4.0.5-1) ...
root@fentlinux:~/home/fent# apt-get install abiword
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes extras:
abiword-common aspell-en libenchant1
Paquetes sugeridos:
abiword-plugins abiword-plugins-gnome abiword-doc aspell-bin
Paquetes recomendados:
abiword-help
Se instalarán los siguientes paquetes adicionales:
abiword abiword-common aspell-en
0 actualizados, 4 se instalarán, 0 no actualizados.
Necesito descargar 5306kB de archivos y se utilizarán 15,7MB de espacio de disco después de desempaquetar.
¿Desea continuar? [S/n]
```

apt-get install lista_de_paquetes

Instala los programas deseados.

apt-get --reinstall install paquete

Reinstala un programa.

apt-get upgrade

Actualiza el sistema.

apt-get dist-upgrade

Actualización más profunda.

apt-get remove lista_de_paquetes

Desinstala un paquete.

apt-get --purge remove lista_de_paquetes

Desinstala un paquete y elimina los archivos de configuración.

apt-get -f install

Para resolver dependencias.



`apt-get clean`

Para limpiar los paquetes descargados e instalados.

`apt-get autoclean`

Para limpiar los paquetes viejos que ya no se usan.

`apt-cache search nombre paquete`

Para buscar un paquete determinado.

`apt-get source paquete`

Descarga archivos fuente.

`apt-get build-dep paquete`

Configura las dependencias de construcción para paquetes fuente.

`apt-get dselect-upgrade`

Sigue las selecciones de dselect.

`apt-show-versions (-u)`

Ideal para mezclar y conocer que paquetes hay instalados, y actualizaciones selectivas.

`apt-cache show paquete`

Obtener más información de un paquete específico.

(da información de tanto paquete instalado como del que no)

`apt-cache showpkg paquete`

Más información aún.

`apt-cache depends paquete`



Para saber de que paquete depende

`apt-file search filemane`

Para encontrar el nombre de un paquete desde un fichero.

`apt-file list paquete`

Lista el contenido de un paquete.

`apt-file update`

Para mantener al día esta función.

OPCIONES

`apt-get -s`

Simula una acción.

`apt-get -d`

Sólo descarga.

`apt-get -y`

No pregunta y asuma que si a todo.

`apt-get -u`

Muestra paquetes actualizados.

`apt-get -h`

Muestra texto de ayuda.

`apt-get -q`



Salida registrable - sin indicador de progreso.

`apt-get -qq`

Sin salida, excepto si hay errores.

`apt-get -f`

Intenta continuar sí la comprobación de integridad falla.

`apt-get -m`

Intenta continuar si los archivos no son localizables

`apt-get -b`

Construye el paquete fuente después de obtenerlo .

`apt-get -V`

Muestra números de versión detallados.

`apt-get -c=?`

Lee este archivo de configuración.

`apt-get -o=?`

Establece una opción de configuración arbitraria, p. ej.

`apt-get -o dir::cache=/tmp`



NOTAS SOBRE LILO Y GRUB

Por: Grotfang y Paulet

LILO

El archivo de configuración del **lilo** es **/etc/lilo.conf**, y podemos acceder a él mediante nuestro editor de texto favorito, y con privilegios de root. Usaré **kate**:

```
kate /etc/lilo.conf
```

Este es mi resultado:

```
# File generated by DrakX/drakboot  
# WARNING: do not forget to run lilo after modifying this file
```

```
default="linux-smp"  
boot=/dev/hda  
map=/boot/map  
keytable=/boot/es-latin1.klt  
prompt  
nowarn  
timeout=100  
message=/boot/message  
menu-scheme=wb:bw:wb:bw
```

```
image=/boot/vmlinuz  
label="linux"  
root=/dev/hda5  
initrd=/boot/initrd.img
```

```
append="devfs=nomount acpi=ht splash=silent"  
vga=788  
read-only
```

```
image=/boot/vmlinuz  
label="linux-nonfb"  
root=/dev/hda5
```



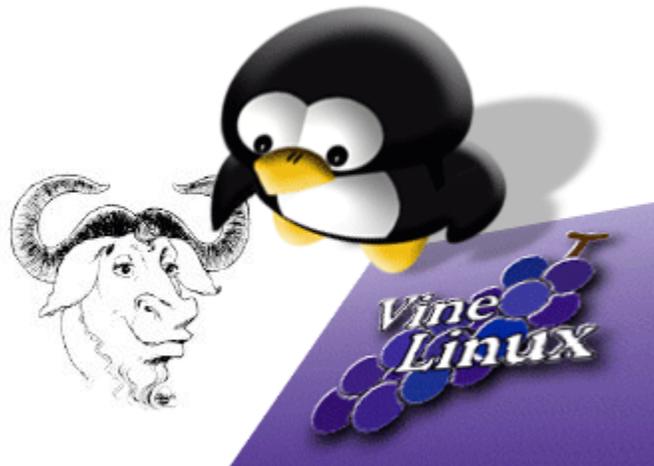
```
initrd=/boot/initrd.img  
append="devfs=nomount acpi=ht"  
read-only
```

```
image=/boot/vmlinuz-2.4.27-0.pre2.1mdk  
label="2427-0pre21"  
root=/dev/hda5  
initrd=/boot/initrd-2.4.27-0.pre2.1mdk.img  
append="devfs=nomount acpi=ht splash=silent"  
vga=788  
read-only
```

```
image=/boot/vmlinuz-smp  
label="linux-smp"  
root=/dev/hda5  
initrd=/boot/initrd-smp.img  
append="devfs=nomount acpi=ht  
splash=silent"  
read-only
```

```
image=/boot/vmlinuz  
label="failsafe"  
root=/dev/hda5  
initrd=/boot/initrd.img  
append="failsafe acpi=ht  
devfs=nomount"  
read-only
```

```
other=/dev/hda1  
label="windows"  
table=/dev/hda
```



Yo tengo instalado en mi ordenador dos sistemas operativos, mandrake y windows. Lo de mandrake no lo adivinarais a simple vista, ya que en ningún sitio lo dice con claridad. He separado mi **lilo.conf** por módulos que he espaciado entre ellos, para poder explicaros más o menos a qué pertenecen. El primer módulo son las opciones de configuración del gestor, que luego os explicaré para que sirve cada una de ellas y como modificarlas. Las cinco siguientes son las correspondientes al mandrake. Y la última de todas es la del otro sistema operativo que tengo instalado... windows.

Paso a explicaros las opciones de configuración más importantes del gestor:



* `default="linux-smp"`: `default` sirve para indicarle a lilo cual será el S.O. que arrancará por defecto, que ahora mismo está configurado para que arranque mandrake. Si queréis que sea windows vuestra primera opción, simplemente sustituir "linux-smp" por "windows", ya que `default` se guía por el nombre que aparece dentro del label de los diferentes módulos.

* `Timeout=100`: Esta opción es la que asigna el tiempo que lilo espera a recibir alguna entrada por parte del usuario, si no la recibe en 10 segundos (se mide en décimas de segundos, 100 es igual a 10 segundos, y 50 igual a 5), se ejecuta `default`, arrancando el S.O. predeterminado. Podéis cambiarlo por el tiempo que creáis oportuno.

* `message=/boot/message`: Esta es la imagen de la pantalla que muestra lilo a la hora de seleccionar el S.O., podéis sustituirla por la que queráis.

* `image=/boot/vmlinuz-smp`: Como podéis ver, los diferentes módulos que se refieren al mandrake tienen ésta línea, lo que hace es indicar el kernel que deben arrancar.

* `Root=/dev/hda5`: Indica la partición raíz, en la que está instalado el mandrake.

* `Other=/dev/hda1`: Y esta nos indica en la que está instalado windows.

* `Label="windows"`: Lo que aparece entre comillas es el nombre con el que aparece en el **lilo**, así que si queremos que en vez de windows, aparezca window\$, pues simplemente tenemos que sustituirlo por lo que aparece entre comillas.

Un último apunte. Cada vez que cambiemos la configuración del kernel, o si instalamos un kernel "a mano" (si lo hacemos mediante actualización no hace falta), debemos introducir el siguiente comando para que los cambios surtan efecto:

```
/sbin/lilo-v-v
```

Diferencias entre GRUB y LILO:

Lilo (LInux LOader) es bastante parecido a grub (GRand Unified Boot Loader), las diferencias más importantes son las siguientes:

- Lilo no puede leer las particiones ext2, a diferencia de grub, que sí las lee.
- Lilo almacena información sobre el kernel o el S.O. que se pueda cargar en el MBR (ésta es la razón por la que hay que introducir el comando que expliqué anteriormente).



GRUB

Las presentes notas han sido redactadas con la intención que os sea de utilidad para aprender a editar y configurar con vuestro estilo personal el **grub**, el objetivo es que lo entendamos para saber que estamos viendo y como es posible modificarlo.

El archivo de configuración de **grub** lo tenéis en el **/boot/grub** y se llama **menu.lst**

Y podéis editarlo perfectamente como **root** con vuestro editor favorito, por ejemplo, desde consola y con **nano**:

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

A continuación reseñaré como modificar el sistema predeterminado, el timeout y el nombre de vuestros sistemas en el **grub**.

* Arranque por la primera entrada por defecto:

Si buscamos en nuestro **menu.lst**, observaremos al principio del todo una entrada como está:

default 0

Esto implica que por defecto el **grub** arrancará la primera o primer sistema que aparezca en el listado, sistema que reconoce el **grub** a todo aquello que de inicio con la palabra **title**, que identificará como el título o nombre del sistema en cuestión.

Si, por ejemplo, modificamos el **default 0** por **default 2**, implicará que arrancará por defecto el tercer sistema o entrada que tengamos establecido en el **grub** y así sucesivamente.

Veamos un ejemplo de un **grub**:

```
default 0
timeout 15
color cyan/blue white/blue
foreground ffffff
background 0639a1
gfxmenu /boot/grub/message
```



```
title MEPIS at sda5, kernel 2.6.10
kernel (hd0,4)/boot/vmlinuz-2.6.10 root=/dev/sda5 nomce psmouse.proto=imps quiet splash=verbose
vga=791
initrd (hd0,4)/boot/initrd.splash
```

```
title MEPIS at sda5, kernel 2.4.29
kernel (hd0,4)/boot/vmlinuz-2.4.29 root=/dev/sda5 nomce quiet splash=verbose vga=791 hda=ide-scsi
hdb=ide-scsi
initrd (hd0,4)/boot/initrd.splash
```

```
title Windows at sda1
rootnoverify (hd0,0)
chainloader +1
```

```
title MEMTEST
kernel /boot/memtest86.bin
```

Arrancaría la primera opción: “title MEPIS at sda5, kernel 2.6.10”

Pero si modificamos el **default 0** por **default 2**, arrancaría de forma predeterminada la opción de windows.

Como es obvio señalar que **grub** empieza a contar por 0 siendo esta la primera entrada de title en el grub, si por ejemplo tuviéramos 15 y quisiéramos que por defecto arrancará la última deberíamos modificar el **grub** colocando un **default 14**.



* Sin uno fijo predeterminado de antemano:

Y si lo que pretendo no es tener ningún sistema por defecto y que esta predeterminación este sujeta al último sistema utilizado, basta con realizar una serie de modificaciones, muy útiles cuando tu idea no es tener un sistema predeterminado y si te guste que en un principio **grub** arranque por defecto el último sistema utilizado.

Para ello editamos el archivo de configuración del **grub**, como ya se ha comentado anteriormente y modificamos el **default 0** por un "**default saved**" arriba del todo.

Ejemplo:

default saved



```
timeout 30
color cyan/blue white/blue
foreground ffffff
background 2f5178
splashimage /boot/grub/mepis.xpm.gz
```

Y debajo de cada SO o versión kernel instalada un "**savedefault**"

Ejemplo:

```
title MEPIS 2004.06 at sda8, kernel 2.6.7
kernel (hd0,7)/boot/vmlinuz-2.6.7 root=/dev/sda8 nomce psmouse.proto=imps quiet splash=verbose
vga=791
initrd (hd0,7)/boot/initrd.mepis
savedefault
```

De esta forma y tras guardar los cambios cada vez que reiniciemos o encendamos el cacharro seleccionará la última entrada utilizada como opción predeterminada.

Veamos mi **grub**, un poco largo, y remarcaré lo comentado a modo de ejemplo, pero es que a estos chicos de **ubuntu** les gusta explicarlo todo, así que optaré por reducirlo y añadir un [...] a aquello que este fuera de explicación o sean comentarios realizados por ubuntu dentro del **grub** facilitando la lectura y comprensión de lo mencionado.

es un comentario que grub no lee.

menu.lst - See: grub(8), info grub,

[...]

default num

[...]

default saved

timeout sec

[...]

timeout **20**

[...]



```
## ## End Default Options ##
```

```
title      Ubuntu, kernel 2.6.10-4-386
root      (hd0,5)
kernel    /boot/vmlinuz-2.6.10-4-386 root=/dev/sda6 ro quiet splash
initrd    /boot/initrd.img-2.6.10-4-386
savedefault
```

```
title      Ubuntu, kernel 2.6.10-4-386 (recovery mode)
root      (hd0,5)
kernel    /boot/vmlinuz-2.6.10-4-386 root=/dev/sda6 ro single
initrd    /boot/initrd.img-2.6.10-4-386
savedefault
```

```
title      Ubuntu, kernel memtest86+
root      (hd0,5)
kernel    /boot/memtest86+.bin
```

```
### END DEBIAN AUTOMAGIC KERNELS LIST
```

```
[...]
```

```
title      Mepis 2.4xxx (on /dev/sda5)
root      (hd0,4)
kernel    /boot/vmlinuz-2.4.29 root=/dev/sda5
savedefault
```

```
[...]
```

```
title      Mepis 2.6xxx (on /dev/sda5)
root      (hd0,4)
kernel    /boot/vmlinuz-2.6.10 root=/dev/sda5
savedefault
```

```
[...]
```

```
title      Windosete
root      (hd0,0)
makeactive
```



chainloader +1
savedefault

Para finalizar haré mención a la opción de variar el tiempo en el que **grub** tarda en escoger la opción predeterminada para arrancar el sistema operativo y como modificar el nombre del sistema operativo en cuestión

Algo que es puede resultar muy útil es modificar el tiempo de arranque establecido por defecto en el **grub**, para unos es poco tiempo, para otros demasiado el que establece vuestro **grub** en vuestra versión linux.

Esto es fácilmente modificable, editamos el **grub** con vuestro editor favorito y modificamos el parámetro que identificaréis por: **timeout**, normalmente es la segunda línea no comentada del **grub**.

Y allí le damos los segundos que deseamos que tarde **grub** en cargar una entrada o sistema:

5, 10, 15, 20, ... para gustos. Como pueden ver en mi **grub**, un poco más arriba, tengo configurado que si en 20 segundos no le he dado una orden contraria inicie sistema con la opción predeterminada.

Como siempre guardamos los cambios.

Y si aún deseáis personalizar el título de vuestro linux o sistema bastaría con modificar el nombre que aparece al lado de **title** por el que más os guste.

Por ejemplo, a windows le llamo windosete, y ahora acabo de reinstalar mis sistemas, y sus nombres están levemente modificados, pero muy pronto empezaré a trastear, meter parches y demás, ante ello deberé añadir nuevas entradas a mi **grub** y nuevos nombres con el que identificarlos y aclararme con el mapa conceptual de mi **grub**.

Por ejemplo, **mepis** lo actualizaré a la última versión del kernel y lo nombraré como title: mepis a la última o mepis gringero y así un largo etc.



Espero allá sido de utilidad y os sirva para tener un **grub** más acorde con vuestro estilo, a la par espero continuar en otro momento reseñando nuevas entradas y funciones del **grub**.



INSTALACIÓN DE UNA EMISORA DE RADIO ONLINE BAJO LINUX

Por: Vl@d

La aparición de internet ha supuesto para muchas tecnologías y medios de comunicación, y proporcionando a los internautas servicios que hasta ahora solo podían obtener de medios convencionales.

Este es el caso de radio. Quien le iba a decir a Marconi que su invención se iba a popularizar tanto y mucho menos que esta se iba a poder obtener de un medio tan fantástico como internet.

Y por supuesto, el sistema operativo del pingüino no iba a ser ajeno a todas las posibilidades que nos ofrece esta gran red informática y prueba de ello es el siguiente manual en el que veremos las herramientas, instalación y puesta en funcionamiento de una emisora online de radio bajo sistema operativo Linux.

Para conseguir nuestro propósito hemos de hacernos con los siguientes programas y librerías:

-Muse ---> Reproductor mediante el cual controlaremos la emisión de las canciones y que “engancharemos” al servidor de radio.

-Icecast ---> Servidor de radio online.

-Lame ---> Necesario para el funcionamiento de Muse.

-Todas aquellas librerías necesarias para la reproducción de archivos mp3 y ogg.

Bien vamos a hacernos con Icecast. Lo encontraremos en esta dirección:

<http://www.icecast.org/download.php>

Y compilamos:

```
tar -zxvf icecast-2.2.0.tar.gz
cd icecast-2.2.0/
./configure
make
su
make install
```



Y ahora el turno de lame, primero descargamos:

http://sourceforge.net/project/showfiles.php?group_id=290

Y a continuación compilamos:

```
tar -zxvf lame-3.96.1.tar.gz
cd lame-3.96.1/
./configure
make
su
make install
```

Muse lo podemos descargar de la siguiente dirección:

<ftp://ftp.dyne.org/muse/releases/>

Desde ahí descargaremos la versión que deseemos y en el formato que queramos, si es un *.rpm*:

```
rpm -i MuSE-0.9-1.i386.rpm
```

Y si es un tarball:

```
tar -zxvf MuSE-0.9.1.tar.gz
cd MuSE
./configure
make
su
make install
```



Bien, ya tenemos las herramientas necesarias, ahora vamos a ponerlas en funcionamiento. Lo primero que haremos es darle un vistazo al archivo de configuración de Icecast, tenemos una completísima descripción de él en la página oficial del proyecto:

http://www.icecast.org/docs/icecast-2.2.0/icecast2_config_file.html

En este archivo vamos a poder especificar pautas de comportamiento y funcionamiento del servidor de radio, pero nosotros solo vamos tener que cambiar de este archivo el sitio donde queremos los logs de



de Icecast.

Para llevar un control adecuado de nuestra radio vamos a crear un subdirectorio en nuestro directorio personal, dentro de él dejaremos las canciones y los logs:

```
[vlad@localhost vlad]$ mkdir radio
```

Ahora como root editaremos el archivo de configuración de Icecast y cambiaremos los logs:

```
root@localhost vlad)# nano /compilados/icecast-2.1.0/conf/icecast.xml.in
```

Localizaremos la linea:

```
<logdir>@localstatedir@/log/@PACKAGE@</logdir>
```

Y la cambiaremos por:

```
<logdir>/home/vlad/radio/</logdir>
```

Por motivos de seguridad no podemos arrancar Icecast como root, así lo haremos como usuario normal de la siguiente forma:

```
[vlad@localhost vlad]$ icecast -b -c /compilados/icecast-2.1.0/conf/icecast.xml.in
```

```
Starting icecast2
```

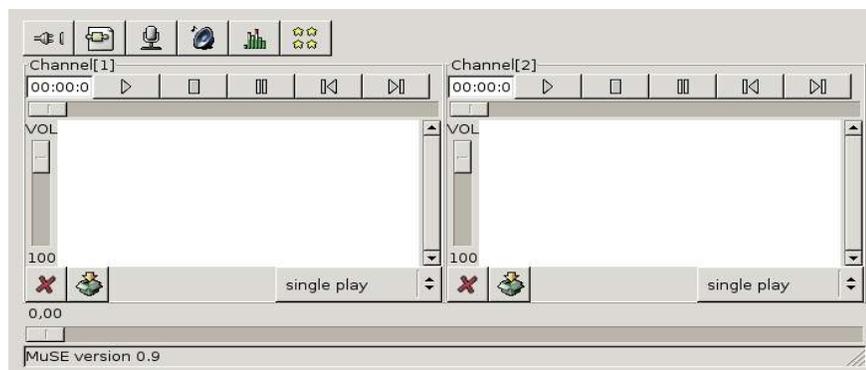
```
Detaching from the console
```

Ya tenemos nuestro servidor de radio funcionando, ahora nos vamos a ir al Muse y vamos a transmitir.

Arrancamos Muse:

```
[vlad@localhost vlad]$ muse
```

Este es su aspecto:





Pinchamos en el primer icono (el del enchufe) y posteriormente una vez abierta la siguiente ventana, pinchamos a su vez en “Avanzado”. Esto es lo que veremos:

The screenshot shows the 'Add Server' dialog box with the following configuration:

- Quality: (low) [slider] (high) Mode: mono
- 2,0 (16Kbit/s 22050Hz)
- bitrate: 16 frequency filtering: auto
- frequency: 22050 lowpass Hz: 0
- highpass Hz: 0
- Record Ogg/Vorbis Stream: I'm Expert...
- Enter filename: [text field] Profile...
- Browse... Record now! Apply

The bottom section of the dialog box contains the following fields:

- Host: localhost Port: 8000 Mnt: /live
- Name: Streaming with MuSE Url: http://muse.dyne.org
- Description: Free Software Multiple Streaming Engine
- Login Type: icecast 2 Pass: [text field]
- Buttons: Connect Profile... Delete

Desde este panel podemos configurar la calidad de nuestra emisión y por supuesto empezar a emitir. Para ello lo primero que hemos de hacer es introducir la contraseña en el campo correspondiente, dicha contraseña es por defecto “hackme”, pero podemos cambiarla editando el archivo de configuración y cambiándola por lo que deseemos.

Ahora nos vamos al panel principal de Muse y añadimos una canción poniéndola a sonar y en el panel de emisión por red pinchamos en “Conectar”. Ya estamos emitiendo, para comprobar que es así abrimos el reproductor xmms y en la lista de reproducción hacemos “Añadir/URL/” y en la ventana ponemos:



<http://localhost:8000/live>

Damos al play y deberíamos escuchar nuestra canción.



IMAGEMAGICK DESDE CONSOLA

Por: ZX80

ImageMagick es una herramienta muy útil de tratado de imágenes a un nivel sencillo, sin las pretensiones de otros grandes como Gimp.

Este programa vale tanto para visualizado como para procesado de imágenes como convertir, rotar, añadir efectos, y viene en todas las distribuciones de Linux.

El uso es muy sencillo, pero cuando un usuario no tiene entorno gráfico la cosa es algo más complicada. Este manual ayudará a estos usuarios en el tratado de imágenes con ImageMagick.

Manipulación básica.

Para la creación de un thumbnail o copia en miniatura para ponerlo en webs o en catálogos usaremos la sintaxis:

```
convert -sample 80x60 zx80.jpg zx80-thumb.jpg
```

Convertirá una imagen llamada zx80.jpg en zx80-thumb.jpg a un tamaño de 80x60. Este tamaño no es real del todo, ya que ImageMagick advertirá que la imagen puede quedar deformada y automáticamente la ajustará, pudiendo quedar de un tamaño tal que 76x58.

También se pueden usar porcentajes:

```
convert -sample 45%x45% zx80.jpg zx80-thumb.jpg.
```

Si a la vez queremos rotar la imagen siguiendo las agujas del reloj pondremos:

```
convert -rotate 90 zx80.png zx80-2.png
```

Si quisiéramos rotar la imagen en contra de las agujas del reloj, pondríamos un número negativo:

```
convert -rotate -90 zx80.png zx80-2.png
```

Para hacer un flip, o lo que es lo mismo, cambiar la parte de arriba de la imagen a abajo o la derecha a la izquierda, teclearemos esta línea:



```
convert -flip zx80.jpg zx80-flip.jpg  
convert -flop zx80.jpg zx80-flop.jpg
```

Si tenemos una colección de imágenes para cambiar de formato, podemos usar el comando mogrify con el modificador -format:

```
mogrify -format jpg *.png
```

También podemos crear thumbnails con mogrify y cambiarle el formato a png, por ejemplo:

```
mogrify -format png -sample 30% $\times$ 30% *.jpg
```

Añadir efectos.

Los efectos especiales en las fotos las dotan de cierto don artístico. Algunos efectos de ImageMagick son el charcoal, colorize, implode y solarize, entre otros. Estos patrones se fijan por medio de numeración para especificar la cantidad de efecto que tendrá la imagen.

En el caso de charcoal, una especie de boceto inicial de la imagen original, pondremos este comando:

```
convert -charcoal 3 zx80.png zx80-coal.png
```

Si quisiéramos poner otro tipo de efectos solo deberemos cambiarlo.

Jugando con PDFs

Si tenemos una colección de fotos, podemos añadirlas a un formato portable de datos (PDF) para catalogarlas o mandarlas a un amigo. El comando mágico que hace esto es:

```
convert *jpg zx80.pdf
```

También se puede extraer imágenes de un pdf:

```
convert zx80.pdf zx80-imagen%04d.png
```





- El modificador %d selecciona el formato de salida. Debe ser un numero decimal.
- Si ponemos un cero (0) nos servirá para el caso en que tengamos mas de 10 fotos, por lo que a las primeras 9 les pondrá el cero delante para estar ordenadas correctamente.
- El 4 indica el número total de dígitos que podrá usar. Por ejemplo 0001, 0002, 0003.... ideal para colecciones muy numerosas.
- Si no se especifica el modificador %d ImageMagick se encargará de poner un solo valor a la imagen.

Añadir texto.

Con la cantidad de imágenes que solemos almacenar, ¿cómo distinguirlas? ¿Y sí pudiéramos añadirle un Tag o cabecera a la imagen?. Pues a eso vamos:

```
convert -font helvetica -fill red -pointsize 30 \  
> -draw 'text 110,320 "Tibidabo, Barcelona"' \  
> zx80.png zx80-tag.png
```

Esto especifica el color, la fuente, el tamaño, la localización y el propio texto.

NOTA: Se puede poner todo en la misma línea de comando, la barra solo indica cambio de línea.

También podríamos poner textos con dos localizaciones diferentes en diferente color así:

```
convert -font helvetica -fill red -pointsize 30 \  
> -draw 'text 110,320 "Tibidabo, Barcelona"' \  
> -fill blue -draw 'text 150,380 Domingo' \  
> zx80.png zx80-text.png
```

Si nos fijamos en las comillas vemos que en la primera frase tenemos dobles comillas (") cuando en la segunda palabra no están puestas. Esto se debe a que si queremos poner más de una palabra deberemos especificarlo con las dobles comillas. Se puede además especificar tamaños diferentes para ambos textos.

NOTA: En algunas distribuciones Linux puede que la ruta de las fuentes no coincida con la ruta de ImageMagick, por lo que se deberá hacer un enlace simbólico o copiar las fuentes a la ruta apropiada, a



no ser que especifiquemos el Path en el propio comando.

Bordes.

Para la aplicación de frames o bordes en una fotografía se puede especificar en modo sólido o algo transparente.

```
convert -bordercolor white -border 5x5 zx80.png zx80-borde.png
```

Para saber los colores disponibles para bordes, fondos y textos tenemos el comando `-list color`.

Otra opción de borde es el raise o borde en 3D trasparente.

```
convert -raise 15x15 zx80.jpg zx80-raised.jpg
```

Para bordes algo más elaborados podéis usar las opciones de `-mattecolor` y `-frame` con los modificadores `-bordercolor` y `-border`.

En fin, este minimanual podría ser más extenso, pero lo más básico se ha visto reflejado y además, siempre contamos con las páginas man y la propia ayuda de convert.

www.fentlinux.com

