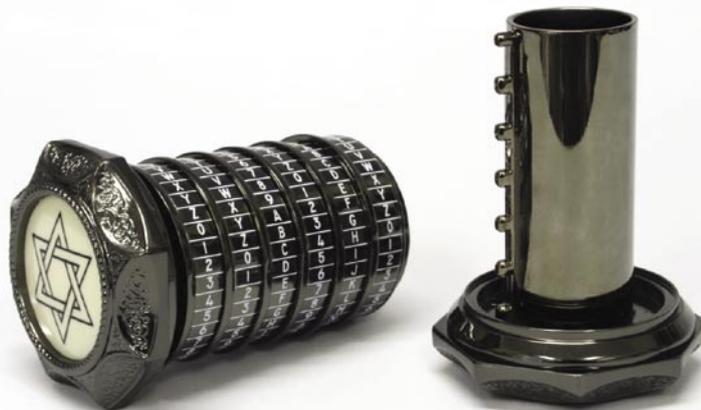


# Creación de una distribución personalizada

Adolfo Lozano Tello, Marcos Blanco Galán, CIRL

A pesar del considerable número de distribuciones Linux existentes actualmente, es posible que se requiera crear una que se adapte a las necesidades específicas de un grupo de usuarios o de los integrantes de una empresa u organización.



linux@software.com.pl

La funcionalidad de una distribución puede buscar más o menos simplicidad, completitud, eficiencia, números de paquetes instalados, tamaño, o aplicaciones específicas que serán útiles, en principio, sólo para el grupo de usuarios para la que fue creada. En este artículo se muestra una guía muy simple para crear una distribución Linux personalizada, adaptando una existente.

## Qué es y para qué sirve una distribución personalizada

Toda distribución Linux está formada por un elemento central, el kernel Linux, al que se añaden una serie de librerías y paquetes de software que proporcionan funcionalidad al sistema. Una de las características principales que diferencia unas distribuciones de otras es, precisamente, el conjunto de aplicaciones o paquetes que la forman. Originalmente, cada distribución incluye sus propios paquetes con el fin de satisfacer las necesidades de un determinado grupo de usuarios. Esta característica da lugar a una clasificación de las distribuciones Linux según el propósito. Las distribuciones de propósito general, por ejemplo, están enfocadas principalmente a

usuarios domésticos e incluyen aplicaciones de uso frecuente para realizar tareas básicas, tales como editar textos, navegar por Internet, ver películas, escuchar música o enviar emails. Algunas de las distribuciones de propósito general más extendidas son Mandriva, openSUSE, Fedora, Debian y Ubuntu. Por otro lado, también existe un grupo de distribuciones más específicas creadas con propósitos determinados y dotadas de software enfocado a aspectos más concretos. Distribuciones de este tipo son, entre otras, Wifislax (auditoría wireless), Linux Multimedia Studio (edición de audio), SuperGamer (entretenimiento) y Damn Small Linux (dirigida a equipos con pocos recursos). El hecho de disponer de un gran número de posibilidades no siempre es motivo suficiente para encontrar la solución idónea de cara a los requerimientos de los usuarios finales. De hecho, al existir tanta variedad, se complica la tarea de elegir la opción más adecuada.

La creación de una distribución Linux personalizada puede estar motivada por varios aspectos. Haciendo alusión a cuestiones técnicas, algunas de las distribuciones existentes hoy día exigen demasiados recursos para ser ejecutadas en equipos antiguos o poco potentes. Otra limitación en este sentido es la posible falta de drivers adecuados para hacer



funcionar determinados componentes hardware de la máquina anfitriona. Estos inconvenientes, al igual que otras cuestiones que comentaremos a continuación, pueden solventarse gracias a la flexibilidad del kernel Linux y de las distribuciones Linux en general. Otro aspecto a tener en cuenta es el conjunto de aplicaciones de la distribución. Es muy difícil encontrar una distribución que contenga todas versiones de aplicaciones y librerías requeridas específicamente para cada caso de uso. Otras distribuciones, en cambio, incorporan gran cantidad de software innecesario del cual se puede prescindir, como habitualmente ocurre en las distribuciones de propósito general. Organizaciones, empresas y otras comunidades de usuarios, además de servir como sello identificativo, encuentran en las distribuciones Linux personalizadas una solución adecuada para cubrir sus necesidades y objetivos concretos. Ejemplo de ello son las distribuciones propias de algunas comunidades autónomas como Linux PYME para gestión empresarial en Extremadura o la distribución de propósito general MoLinux en Castilla-La Mancha (se puede ver un listado de algunas distribuciones Linux en: [http://cir1.unex.es/index.php?seccion=pasoalinux#introduccion\\_2](http://cir1.unex.es/index.php?seccion=pasoalinux#introduccion_2)).

**Listado 1.** Abrimos un terminal y asignamos valores para las variables

```
export WORK=~/.work
export CD=~/.cd
export FORMAT=squashfs
export FS_DIR=casper
```

Las preferencias del usuario final también influyen en la decisión de crear una distribución personalizada. Dentro de estas preferencias se incluyen, por ejemplo: las herramientas para la gestión de paquetes, el entorno gráfico, el tipo de instalador, las licencias de software, la cantidad de paquetes disponible en los repositorios, la frecuencia de actualización de los mismos, etc.

Disponer de una distribución personalizada con instalador propio evita el tener que repetir el tedioso proceso de adaptación y personalización de cada uno de los sistemas en caso de tratarse de varios equipos informáticos. La nueva distribución podría distribuirse cómodamente en CD, DVD o pendrive y, además, añadir la capacidad de ejecutarse en modo Live.

En resumen, existen varias razones por las que realizar una distribución Linux a medida, ya que puede suponer un importante ahorro de tiempo, recursos y mejoras en el rendimiento de realización de diversas actividades.

### Distribuciones instalables en el disco duro y distribuciones Live

Dispuestos a crear un sistema Linux personalizado, el siguiente paso es decidir cómo se distribuirá entre los usuarios. Lo más frecuente a día de hoy es utilizar el CD o DVD como medio principal de distribución, aunque resulta muy útil el almacenamiento en pendrive. La elección del soporte va a depender principalmente del tamaño final del conjunto de paquetes que se desea proveer. La distribución también podría proporcionarse (habitualmente en un servidor

web para ser descargada) en formato de imagen ISO para su posterior grabación en un soporte de almacenamiento.

Una opción interesante según la finalidad de la distribución es considerar la posibilidad de ejecutar el nuevo sistema Linux en modo Live, es decir, de forma autónoma y sin la necesidad de ser instalada en el disco duro. Llegados a este punto y de cara a apartados que se verán más adelante, es conveniente describir qué es y cómo funciona una distribución Live. La característica principal de toda distribución Live es la capacidad de ejecutarse de forma autónoma, sin necesidad instalar nada o alterar el estado actual del disco duro. Un Live CD o Live DVD es, básicamente, una instalación típica de Linux en soporte magnético, similar a la instalación que podría realizarse sobre un disco duro corriente.

### Diferencias entre una instalación típica de Linux y un sistema Live

Una de las diferencias más destacadas entre una instalación típica y una distribución Live reside en que el soporte donde se encuentra la distribución Live es únicamente de lectura. Sin embargo, Linux también necesita escribir en determinadas partes del sistema para funcionar correctamente. Esto se puede solventar utilizando UnionFS, un servicio cuyo resultado final es la posibilidad de escribir en cualquier parte del sistema. UnionFS permite montar un sistema de archivos formado por la unión de otros sistemas de archivos. En este caso, el directorio raíz del sistema de sólo lectura del disco y el sistema de

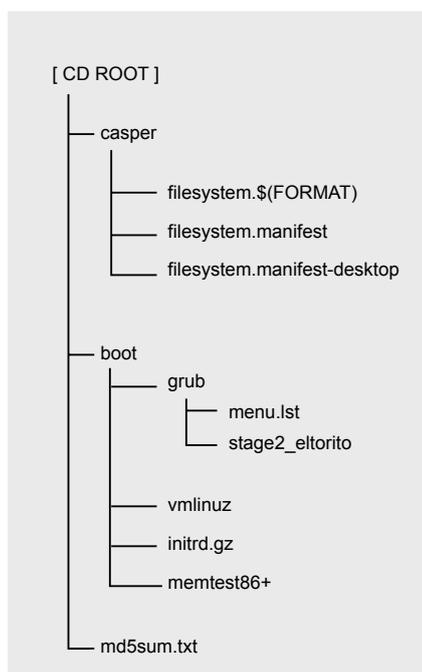


Figura 1. Estructura interna del Live CD/DVD

**Listado 2.** Copiar el sistema base personalizado al directorio de trabajo

```
sudo rsync -av --one-file-system --exclude=/proc/* --exclude=/dev/*\
--exclude=/sys/* --exclude=/tmp/* --exclude=/home/*\
--exclude=/lost+found / ${WORK}/rootfs
```

**Listado 3.** Montando /dev, /proc y /sys

```
sudo mount -o bind /dev/ ${WORK}/rootfs/dev
sudo mount -t proc proc ${WORK}/rootfs/proc
sudo mount -t sys -o bind /sys ${WORK}/rootfs/sys
```

**Listado 4.** Eliminando archivos innecesarios

```
# for i in "/etc/hosts /etc/hostname /etc/resolv.conf /etc/timezone /etc/
fstab /etc/mtab
/etc/shadow /etc/shadow- /etc/gshadow /etc/gshadow- /etc/gdm/gdm-
cdd.conf /etc/gdm/gdm.conf-
custom /etc/X11/xorg.conf /boot/grub/menu.lst /boot/grub/device.map"
do
    rm $i
done 2>/dev/null
```

archivos de almacenamiento temporal (`tmpfs`) residente en memoria.

Una segunda diferencia entre una instalación típica y un sistema Live está relacionada con la ubicación de la raíz del sistema de archivos. En una instalación típica en disco duro, la ubicación de la raíz del sistema de archivos es fija.

En contraposición, el dispositivo de arranque no siempre es el mismo cuando se trata de distribuciones Live, ya que podrían existir

varias unidades CD-ROM y, además, cada una de ellas disponer de interfaces de conexión diferentes (`ide`, `sata`, `scsi`, etc). En definitiva, para que el sistema de archivos pueda montarse correctamente en el modo Live, ha de existir un mecanismo capaz de identificar la raíz del dispositivo y, acto seguido, cargar los módulos del kernel adecuados para acceder al CD-ROM y al sistema de archivos del CD.

Para que el sistema de archivos no tenga problemas de espacio a la hora de ubicarlo en

el CD o DVD, habitualmente éste suele comprimirse utilizando lo que se conoce con el nombre de SquashFS, un sistema de archivos comprimido de sólo lectura.

Puesto que será necesario autodetectar el tipo de sistema de archivos SquashFS durante el arranque del sistema Live, éste también deberá disponer de los módulos adecuados para montarlo.

### Arranque de un sistema Live

Con el propósito de ultimar los preparativos que caracterizan a un sistema Live vistos anteriormente, éste utiliza lo que se conoce con el nombre de *Intramfs* o *Initial Root Filesystem*. Básicamente, *intramfs* contiene un shell básico, los módulos del kernel necesarios para montar el sistema de archivos raíz y una serie de scripts para realizar ciertas tareas durante el arranque en modo Live. El script más importante de *intramfs* es uno llamado `init`, ubicado en la raíz del *intramfs*.

Antes de continuar, debe definirse el concepto de gestor de arranque. El gestor de arranque es un programa sencillo diseñado para preparar, como su propio nombre indica, el arranque de un sistema operativo. La ejecución del gestor de arranque o también llamado *bootloader* tiene lugar una vez que la BIOS (*Basic Input-Output System*) ha reconocido el hardware tras encender el equipo. El gestor de arranque es el responsable de cargar el kernel Linux e *intramfs* en memoria principal.

A continuación, comienza la ejecución del kernel. El kernel desempaqueta *intramfs* y éste es montado como sistema de archivos raíz inicial.

A continuación, tiene lugar la búsqueda e invocación del script `init`. Una de las funciones más importantes del script `init` es encontrar y montar el verdadero sistema de archivos raíz. Cualquier otra operación especial requerida durante el arranque del sistema Live también podría incluirse en los scripts de arranque de *intramfs*.

*Intramfs* puede generarse manualmente, sin embargo, es posible automatizar este proceso gracias a herramientas como `update-intramfs`. Los scripts personalizados que habilitan el arranque en modo Live son copiados a rutas específicas (`/usr/share/initramfs/scripts`). A continuación, para generar o actualizar *intramfs*, bastaría con ejecutar el comando `update-intramfs`.

De hecho, tampoco es necesario escribir los scripts de *intramfs* a mano, pues existen paquetes con scripts adaptados para facilitar el arranque en modo Live. Un ejemplo de este tipo de paquetes es Casper. Al instalar Casper,



Figura 2. Distribución UEX Linux para la Universidad de Extremadura

#### Listado 5. Eliminando los usuarios que no son del sistema

```
# do
    uid=`cat /etc/passwd | grep "^${i}:" | awk -F":" '{print $3}'`
    [ "$uid" -gt "999" -a "$uid" -ne "65534" ] && userdel --force ${i}
2>/dev/null
done
```

#### Listado 6. Creando ficheros de log

```
# for i in dpkg.log lastlog mail.log syslog auth.log daemon.log faillog
lpr.log mail.warn
    user.log boot debug mail.err messages wtmp bootstrap.log dmesg kern.log
mail.info
do
    touch /var/log/${i}
done
```

#### Listado 7. Copiando las imágenes del kernel

```
sudo cp -vp ${WORK}/rootfs/boot/vmlinuz-${uname -r} ${CD}/boot/vmlinuz
sudo cp -vp ${WORK}/rootfs/boot/initrd.img-${uname -r} ${CD}/boot/initrd.gz
sudo cp -vp ${WORK}/rootfs/boot/memtest86+.bin ${CD}/boot
```



automáticamente se colocan todos los scripts requeridos en el lugar adecuado.

Finalmente, sólo habría que regenerar `initramfs` ejecutando:

```
update-initramfs.
```

## Estructura interna de un Live CD o Live DVD

Este apartado tiene como objetivo describir la estructura interna genérica característica de una distribución Live. El siguiente diagrama mostrado en la figura 1 presenta la jerarquía de directorios:

- `/casper/filesystem.${FORMAT}`: Habitualmente, se trata de un archivo con extensión `.squashfs` que contiene el sistema de archivos comprimido. Si el usuario decidiese instalar el sistema operativo Linux desde el modo Live, el sistema de archivos SquashFS sería descomprimido y copiado al disco duro durante el proceso de instalación.
- `/casper/filesystem.manifest` y `/casper/filesystem.manifest-desktop`: Son dos ficheros de texto utilizados cuando la distribución Live incluye instalador. Cada uno contiene un listado de paquetes. Al finalizar el proceso de instalación, el instalador procede a eliminar los paquetes que aparecen exclusivamente en el fichero `filesystem.manifest`. La finalidad es desinstalar automáticamente los paquetes que no son de utilidad en el sistema recién instalado y, sin embargo, se encontraban instalados en el sistema de archivos SquashFS inicial. Casper y el propio instalador de la distribución son dos ejemplos de paquetes innecesarios que pueden eliminarse de forma segura.
- `/boot/grub/menu.lst`: Fichero de texto plano con las opciones de arranque del modo Live. Es muy común utilizar GRUB (GRand Unified Bootloader) como gestor de arranque.
- `/boot/grub/stage2_eltorito`: Es el bootloader del CD o DVD. Una vez que recibe el control, presenta al usuario el menú principal de GRUB con las opciones indicadas en el fichero de configuración `menu.lst`. La especificación de *cd autoarrancable El Torito* es una extensión de la especificación ISO-9660 diseñada para habilitar el arranque desde un CD o DVD.
- `/boot/vmlinuz`: Kernel Linux comprimido que se cargará en memoria principal con el fin de iniciar el sistema operativo y proveer el entorno de ejecución necesario para correr las aplicaciones.

`/boot/initrd.gz`: Fichero `initramfs` personalizado que habilita el arranque en modo Live. Tanto la imagen del kernel (`vmlinuz`) como el fichero `initramfs` (`initrd.gz`) son referenciados en las opciones de arranque del fichero `menu.lst`.

- `/boot/memtest86+` (opcional): `Memtest86+` es una herramienta utilizada en el diagnóstico del estado actual de la memoria RAM. En la distribución Live es provista como modo de arranque adicional accesible desde el menú de opciones de GRUB.

`/md5sum.txt` (opcional): Fichero de texto plano con los checksums de cada uno de los archivos contenidos en el CD o DVD. En este caso concreto, los checksums o resúmenes MD5 se utilizan como mecanismo para garantizar que los archivos no han sido alterados.

personalizada, instalable, redistribuible, en formato DVD y con la capacidad de ejecutarse en modo Live. Para ello, se parte de un sistema base preinstalado que, posteriormente, será adaptado acorde a la finalidad de la distribución.

En esta guía se ha decidido tomar Ubuntu Linux como sistema base, una distribución de propósito general ampliamente extendida, orientada a usuarios de escritorio y caracterizada por su facilidad de uso y buen soporte hardware. El hecho de utilizar Ubuntu no quiere decir que no sea posible utilizar otras distribuciones como punto de partida. Simplemente, hay que decantarse por una y, en este caso, hemos escogido Ubuntu.

Uno de los motivos que convierten a Ubuntu en una distribución adecuada para ser utilizada como sistema base es la gran cantidad y variedad de software disponible en los repositorios oficiales. Sea cual sea la distribución Linux inicial, esta característica va a permitir minimizar el número de aplicaciones que inevitablemente deberán instalarse de forma manual ante la imposibilidad de utilizar herramientas

## Un ejemplo de creación de una distribución Linux personalizada

En esta sección se muestra un ejemplo práctico de cómo crear una distribución Linux perso-

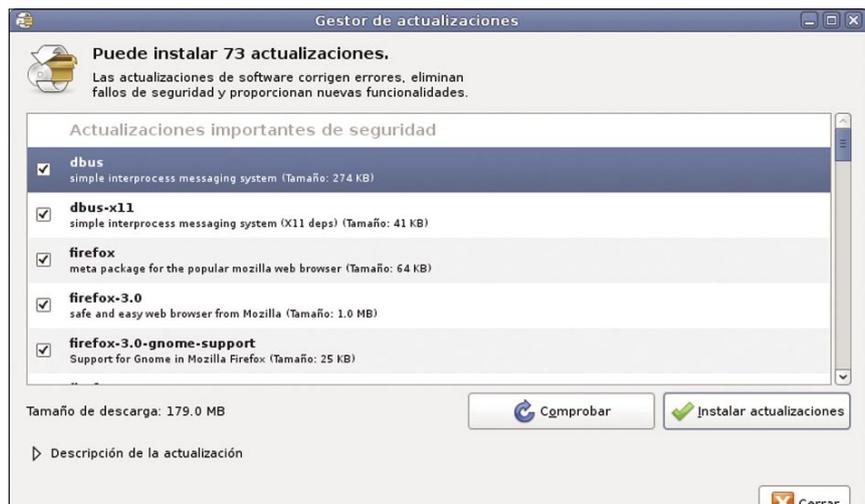


Figura 3. Actualización del sistema base Ubuntu

```
-<gconf>
- <dir name="desktop">
- <dir name="gnome">
+ <dir name="url-handlers"></dir>
+ <dir name="peripherals"></dir>
+ <dir name="applications"></dir>
+ <dir name="font_rendering"></dir>
+ <dir name="sound"></dir>
+ <dir name="interface"></dir>
- <dir name="background">
+ <entry name="picture_options" mtime="1224671884" type="string"></entry>
- <entry name="picture_filename" mtime="1224671884" type="string">
<stringvalue>/usr/share/backgrounds/mi_nuevo_fondo.png</stringvalue>
</entry>
+ <entry name="color_shading_type" mtime="1224671884" type="string"></entry>
+ <entry name="secondary_color" mtime="1224671884" type="string"></entry>
+ <entry name="primary_color" mtime="1224671884" type="string"></entry>
</dir>
</dir>
+ <dir name="apps"></dir>
</gconf>
```

Figura 4. Personalización del fondo de escritorio

automáticas para la gestión de paquetes. Como veremos más adelante, con Ubuntu también va a resultar sencillo añadir la capacidad de ejecución en modo Live y la posibilidad de instalar la distribución en el disco duro, esto último gracias a Ubiquity, un instalador gráfico diseñado especialmente para sistemas basados en Debian y Ubuntu. Es cierto que existen herramientas que facilitan y automatizan el proceso a la hora de adaptar o remasterizar una distribución Linux (Reconstructor, Remastersys o UCK son sólo algunos ejemplos). En esta guía, por el contrario, se ha preferido recurrir exclusivamente a la línea de comandos con el propósito de entender mejor los pasos que se van a llevar a cabo. Por otra parte, aunque suponga una mayor comodidad, en ocasiones, el uso de herramientas externas limita las características que pueden personalizarse durante la construcción de la nueva distribución. Se debe mencionar que el procedimiento descrito en esta guía ha sido la base de UEX Linux alpha (figura 2), un caso real de distribución Linux creada por el

Centro Internacional de Referencia Linux para los alumnos de la Universidad de Extremadura y que actualmente se encuentra en fase de desarrollo de la versión 1.0. Toda la información relativa a UEX Linux está disponible en: <http://ciril.unex.es/uexlinux/index.htm>.

### Requerimientos previos

Antes de comenzar a crear la nueva distribución, hay que asegurarse de disponer del espacio libre mínimo equivalente a dos instalaciones de Ubuntu en disco duro (2 x 4GB), más un espacio adicional suficiente teniendo en cuenta las aplicaciones extra que se van a añadir al sistema base. Puesto que el resultado final va a ser la obtención de una imagen ISO de DVD, es muy recomendable tener una grabadora de DVD para hacer pruebas.

### Instalación y actualización del sistema base

Se parte de una instalación limpia de Ubuntu. En realidad, sería posible utilizar un sistema

Ubuntu existente, aunque no es muy recomendable dado que podrían surgir problemas de dependencias más adelante. Para la elaboración de esta guía se ha utilizado Ubuntu 8.04 LTS (Hardy Heron) de 32 bits. La imagen ISO de Ubuntu puede descargarse gratuitamente desde la web oficial de Ubuntu: <http://www.ubuntu.com>.

Una vez instalado el sistema base, el siguiente paso es actualizar todos los paquetes con las últimas versiones disponibles en los repositorios oficiales. La forma más cómoda de hacer esto es a través del gestor de actualizaciones, una herramienta accesible desde la barra principal de Ubuntu, opción *Sistema*, menú *Administración* (figura 3).

### Personalización del sistema

Hasta el momento no hemos hecho nada fuera de lo normal, únicamente se ha instalado una distribución Linux de propósito general y acto seguido se han actualizado todos sus componentes. Ahora es cuando deberás adaptar el sistema Linux recién instalado a las necesidades, gustos y propósitos específicos.

Las modificaciones del sistema pueden ser múltiples y variadas según los requerimientos de cada caso de uso. Podría ser necesario instalar nuevos paquetes, compilar aplicaciones que no se encontrasen en los repositorios, eliminar programas y servicios innecesarios, realizar cambios en la apariencia (escritorio, temas de ventanas, menús, sonidos, iconos), etc.

En el caso particular de UEX Linux, por ejemplo, hemos recopilado e instalado herramientas y aplicaciones usadas en la parte práctica de algunas de las asignaturas impartidas en las titulaciones de Informática de la Universidad de Extremadura.

Puesto que algunas de estas aplicaciones no estaban en los repositorios, tuvo que recurrirse a la compilación e instalación manual de las mismas. Otros cambios relevantes en UEX Linux respecto al sistema inicial fueron la desinstalación del servicio PulseAudio, la desactivación de Tracker, el uso de OpenJDK como alternativa libre de máquina virtual Java y algunos cambios menores en la interfaz de usuario.

Volviendo otra vez al caso genérico, establecer el conjunto de aplicaciones que se desea proveer es sencillo gracias al uso de las herramientas de gestión de paquetes. Sin embargo, a la hora de modificar aspectos relacionados con la apariencia o la configuración inicial de las cuentas de usuario sí que se necesita saber algo más.

Si, por ejemplo, cambiásemos el fondo de escritorio siguiendo el procedimiento habitual, lo estaríamos haciendo exclusivamente para la

#### Listado 8. Creando filesystem.manifest y filesystem.manifest-desktop

```
sudo chroot ${WORK}/rootfs dpkg-query -W --showformat='${Package}
${Version}\n'
| sudo tee ${CD}/${FS_DIR}/filesystem.manifest

sudo cp -v ${CD}/${FS_DIR}/filesystem.manifest{,-desktop}

REMOVE='ubiquity casper user-setup discover1 xresprobe os-prober
libdebian-installer4'
for i in $REMOVE
do
    sudo sed -i "/${i}/d" ${CD}/${FS_DIR}/filesystem.manifest-desktop
done
```



Figura 5. Usplash de UEX Linux 2008 Alpha



cuenta de usuario actual y no para el resto de cuentas existentes o las que pudieran crearse en un futuro. A continuación, nos centraremos en cómo modificar algunos aspectos relacionados con la apariencia y la configuración inicial de las cuentas de usuario teniendo en cuenta que Ubuntu es el sistema base.

Comenzaremos, por ejemplo, cambiando el fondo de escritorio. Las imágenes de fondo de escritorio se almacenan en `/usr/share/backgrounds/`, lugar donde se ubicará la nueva imagen que deseamos establecer como fondo por defecto. En el fichero `/var/lib/gconf/debian.defaults/%gconf-tree.xml` hay que modificar el parámetro de configuración correspondiente:

- A partir del nodo raíz `gconf`, siguiendo los nodos `desktop`, `gnome`, `background`, llegamos a `picture_filename`. Cambiaremos el valor de la propiedad entre las etiquetas `<stringvalue>` y `</stringvalue>` por la ruta completa del nuevo fondo de escritorio en formato PNG ubicado en `/usr/share/backgrounds/`.

Pasemos a personalizar la pantalla de carga del sistema, también conocida como USplash (ejemplo en figura 5). Vamos a crear un nuevo tema USplash de forma rápida y sencilla a partir de un tema existente que después modificaremos a nuestro gusto. El código fuente del tema original de Ubuntu Hardy Heron puede obtenerse en formato `tar.gz` desde Launchpad: <https://launchpad.net/ubuntu/hardy/+source/usplash-theme-ubuntu/>.

Una vez descargado y descomprimido el fichero `tar.gz`, sólo hay que modificar las imágenes correspondientes según la resolución usada durante el arranque. También sería posible cambiar las imágenes de la barra de progreso `throbber_fore.png` y `throbber_back.png`. Hay que tener en cuenta que todas las imágenes deben crearse o modificarse en base a un esquema de color indexado con una paleta de 256 colores como máximo. Para compilar el nuevo tema es requisito indispensable tener instalado el paquete `libusplash-dev`.

A continuación, compilaremos el tema ejecutando el comando `make` sin ningún parámetro desde el directorio del tema. Como resultado, obtendremos un fichero binario con extensión `.so`. Finalmente, utilizaremos StartUp-Manager para activar el nuevo tema. StartUp-Manager es una aplicación provista por el paquete `startupmanager`, disponible en los repositorios oficiales de Ubuntu.

También podemos cambiar el tema de GDM (GNOME Display Manager). Al igual

que hicimos con USplash, sería más sencillo si modificásemos un tema GDM existente. En UEX Linux (como se ve en la figura 6), por ejemplo, se partió de un tema llamado *Hardy-Simple Colors GDM v2.5*. Éste y otros temas con licencia GPL pueden descargarse en formato `tar.gz` desde *Gnome-Look.org*. Una vez realizados los cambios oportunos, nos dirigiremos a *Sistema > Administración > Ventana de entrada* y, desde la pestaña *Local*, activaremos nuestro tema.

Por último, editaremos el fichero de configuración `/etc/gdm/gdm.conf`, buscaremos la variable `GraphicalTheme` y asignaremos como valor el nombre del nuevo tema.

Si queremos personalizar el tema del escritorio Gnome, copiaremos el nuevo tema en la ruta `/usr/share/themes/`.

A continuación, en el fichero `/var/lib/gconf/debian.defaults/%gconf-tree.xml`, nos dirigiremos al nodo `interface` y modificaremos los valores de las entradas `gtk_theme`, `icon_theme` y `gtk_color_scheme` con los valores asignados en el fichero `.theme` del nuevo

tema para las variables `GtkTheme`, `IconTheme` y `GtkColorScheme` respectivamente.

El tema de ventanas GTK referenciado debe existir en `/usr/share/themes/`, mientras que el tema de iconos ha de ubicarse en `/usr/share/icons/`. Todos estos recursos pueden descargarse de páginas como *Gnome-Look.org* y *Art.gnome.org*. La distribución UEX Linux utiliza un tema de iconos con licencia GPL llamado DeltaGreen, el cuál ha sido ligeramente modificado (figura 7).

Para modificar otras características relacionadas con la interfaz y su comportamiento, recomendamos ver detenidamente los parámetros que admiten los ficheros de configuración de GDM `/etc/gdm/gdm.conf` y el fichero de configuración de Gnome `/var/lib/gconf/debian.defaults/%gconf-tree.xml`.

Respecto a la configuración inicial de las cuentas de usuario, el directorio `/etc/skel/` proporciona una forma segura para hacer que todos los nuevos usuarios partan de la misma configuración cuando son creados con el comando `useradd`.

Listado 9. El contenido de `menu.lst` ha de asemejarse

```
# By default, boot the first entry.
default 0

# Boot automatically after 10 secs.
timeout 10

color black/white blue/white
splashimage=/boot/tux.xpm.gz

title          Iniciar UEX Linux 2008 (Alpha 1) en Modo Gráfico
kernel         /boot/vmlinuz boot=casper quiet splash
initrd         /boot/initrd.gz

title          Iniciar UEX Linux 2008 (Alpha 1) en Modo Gráfico Seguro
kernel         /boot/vmlinuz boot=casper xforcevesa quiet splash
initrd         /boot/initrd.gz

title          Instalar UEX Linux 2008 (Alpha 1)
kernel         /boot/vmlinuz boot=casper only-ubiquity quiet splash
initrd         /boot/initrd.gz

title          Comprobar la integridad del DVD
kernel         /casper/vmlinuz boot=casper integrity-check quiet splash
initrd         /casper/initrd.gz

title          Realizar test de memoria
kernel         /boot/memtest86+.bin

title          Arrancar desde el primer disco duro
root           (hd0)
chainloader +1
```

Simplemente, hay que copiar en dicho directorio aquellos ficheros que queramos incluir en el directorio `/home` de cada usuario. Estos ficheros pueden ser ficheros de configuración de aplicaciones (`.bash_profile`, `.bashrc`, `.bash_login`, `.bash_logout`, `.vimrc`, etc) o cualquier otro tipo de contenido (pdf, odt, avi, etc). Por otra parte, nosotros no vamos a tener que preocuparnos de añadir nuevos usuarios con el comando `useradd`, ya que el usuario de la sesión Live se crea automáticamente y, en caso de instalar la distribución en disco duro, el instalador Ubiquity se encargaría de hacerlo.

En este apartado hemos visto cómo personalizar algunas características básicas con

vistas al nuevo sistema. En la práctica, este proceso podría no ser tan sencillo y requerir más o menos esfuerzo dependiendo de las necesidades de cada distribución.

Una vez finalizada esta fase, el paso siguiente es la remasterización.

### Preparación del entorno de trabajo y la remasterización

Remasterizar una distribución Linux comprende un conjunto de acciones cuyo objetivo final es la creación de una distribución derivada a partir del sistema base que acaba de ser adaptado con los paquetes software y configuraciones según el propósito específico.

Para nuestro ejemplo, el producto de la remasterización va a ser una distribución Linux personalizada basada en Ubuntu, instalable y con capacidad para ejecutarse en modo Live.

A continuación veamos cómo preparar el entorno de trabajo. En primer lugar, abrimos un terminal y asignamos valores para las variables mostradas en el Listado 1.

`WORK` es el directorio donde residen los archivos temporales y el punto de montaje. `CD` es el directorio donde se encuentra el árbol de directorios del CD o DVD. `FORMAT` es el tipo de sistema de archivos; en esta guía utilizaremos `squashfs`. `FS_DIR` es la ubicación real de la imagen del sistema de archivos en el árbol de directorios. Los directorios `WORK` y `CD` se encuentran en el directorio de trabajo del usuario actual (`~`), sin embargo, estos parámetros pueden modificarse sin ningún problema.

A continuación, creamos la estructura de directorios en `WORK` y `CD` con el comando `mkdir`:

```
sudo mkdir -p ${CD}/${FS_DIR},boot/grub) ${WORK}/rootfs
```

Instalamos algunos paquetes que van a ser necesarios en la remasterización:

```
sudo apt-get update
sudo apt-get install genisoimage grub
squashfs-tools linux-ubuntu-modules-
$(uname -r)
```

El paquete `genisoimage` proporciona la herramienta `mkisofs` para generar la imagen ISO-9660 arrancable con la extensión *El Torito*. El paquete `grub` corresponde al gestor de arranque.

El paquete `squashfs-tools` contiene las herramientas necesarias para crear sistemas de archivos SquashFS. El paquete `linux-ubuntu-modules` seguido de la versión del kernel, provee los módulos del kernel cuya versión corresponde con la salida del comando `uname -r`.

### Copia del sistema base personalizado

El siguiente paso es copiar el sistema base personalizado al directorio de trabajo. Para ello, hemos decidido utilizar el comando `rsync` en lugar del tradicional `cp` (Listado 2).

La principal ventaja de `rsync` respecto a `cp` es la posibilidad de excluir ciertos directorios especiales que no han de copiarse. La lista completa con los parámetros que admite `rsync` puede consultarse en el manual de ayuda ejecutando `man rsync` desde un terminal. A



Figura 6. GDM en UEX Linux 2008 Alpha 1

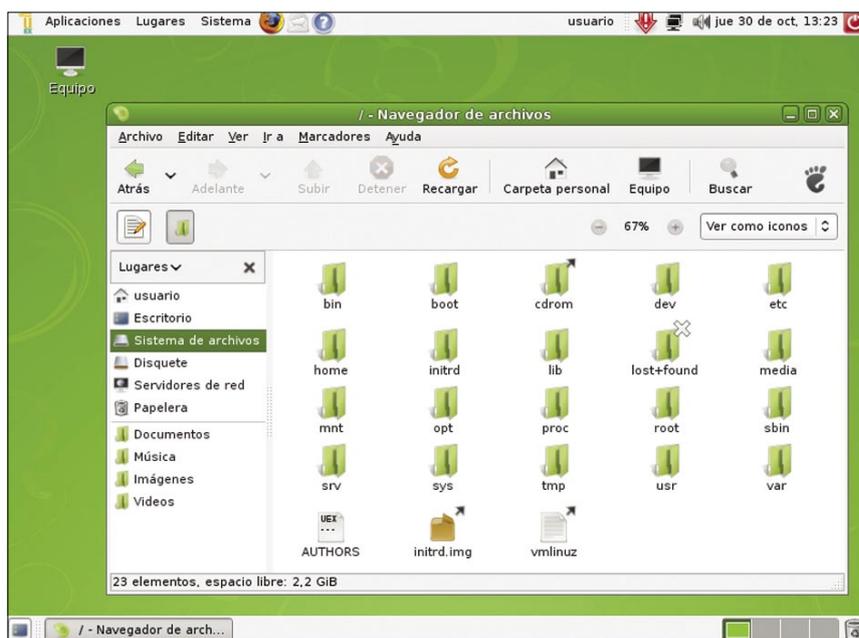


Figura 7. Escritorio Gnome de UEX Linux 2008 Alpha 1



partir de ahora, nos referiremos al sistema base personalizado como *nuevo sistema*.

## Modificaciones adicionales y chroot

El comando `chroot` sirve para crear lo que en Unix y Linux se conoce con el nombre de *jaula*.

Básicamente, una jaula consiste en modificar el directorio raíz del sistema de archivos, afectando únicamente al proceso actual y a sus hijos.

Llegados a este punto, *enjaularemos* el futuro sistema de archivos con el fin de realizar modificaciones adicionales, tales como la preparación del modo Live con Casper, la instalación Ubiquntiy o la limpieza de archivos innecesarios. Mientras nos encontremos dentro de la jaula, los comandos ejecutados afectarán exclusivamente a dicho sistema de ficheros enjaulado. Antes de crear la jaula, es necesario montar `/dev`, `/proc` y `/sys` (Listado 3). Creamos la jaula e invocamos al intérprete de comandos Bash:

```
sudo chroot ${WORK}/rootfs /bin/bash
```

Dentro de la jaula, limpiamos el valor de la variable de entorno LANG: `# LANG=`

Instalamos algunos paquetes esenciales para habilitar el arranque en modo Live:

```
# apt-get update
# apt-get install casper discover1
xresprobe
```

El paquete `casper` proporciona los scripts necesarios para que un sistema basado en `Initramfs` pueda ejecutarse en modo Live.

Para cambiar el nombre de usuario y del equipo durante la sesión Live, modificaremos las variables correspondientes en `/etc/casper.conf`.

El paquete `discover1` provee Discover, un sistema de identificación de hardware que durante el arranque recorre los buses y carga los controladores apropiados.

El paquete `xresprobe` habilita el sondeo de resoluciones del servidor X, tanto en pantallas de equipos portátiles como en otras compatibles con el estándar DCC.

Ubiquntiy va a ser el instalador del sistema Live, por tanto, habrá añadir el paquete correspondiente:

```
# apt-get install ubiquity
```

La instalación de Casper requiere actualizar `initramfs`, aunque no sin antes haber regenerado la lista de dependencias de todos los módulos:

```
# depmod -a $(uname -r)
# update-initramfs -u -k $(uname -r)
```

A continuación, eliminamos archivos innecesarios (Listado 4). Nota: Con `>` lo que hacemos es redirigir la salida `stderr` (*standard error output*). Limpiamos la caché de APT (`/var/cache/apt/archives/`):

```
# apt-get clean
```

También eliminamos los siguientes archivos y directorios:

```
# rm -r /tmp/* /root/* 2>/dev/null
# rm /boot/*.bak 2>/dev/null
```

Eliminamos los usuarios que no son del sistema, es decir, aquellos cuyo ID es superior a 999 (Listado 5). Seguimos limpiando el entorno `chroot`:

```
# find /var/run /var/log /var/mail /var/spool /var/lock /var/backups /var/tmp -type f -exec rm {} \;
```

Si utilizamos GDM, volveremos a crear el fichero de configuración `/etc/gdm/gdm.conf`, ya que una configuración distinta podría interferir en el login automático del Live CD:

```
# [ -f "/etc/gdm/factory-gdm.conf" ]
&& cp
-f /etc/gdm/factory-gdm.conf
/etc/gdm/gdm.conf
2>/dev/null
```

Crearemos algunos ficheros de log que anteriormente ya fueron eliminados.



Figura 8. Logo de CirL

### Listado 10. Generando la imagen ISO-9660

```
sudo mkisofs -b boot/grub/stage2_eltorito -no-emul-boot -boot-load-size 4
-boot-info-table -V
#####
"UEX_Linux_2008_Alpha1" -cache-inodes -r -J -l -o ~/uexlinux2008a1-livedvd
.iso $CD
```



### En la red

- Centro Internacional de Referencia Linux (CIRL): <http://ciril.unex.es>
- Distribución UEX Linux del Centro Internacional de Referencia Linux (CIRL): <http://ciril.unex.es/uexlinux/index.htm>
- Manual de instalación estándar de Ubuntu: [http://www.guia-ubuntu.org/index.php?title=Instalaci%C3%B3n\\_est%C3%A1ndar](http://www.guia-ubuntu.org/index.php?title=Instalaci%C3%B3n_est%C3%A1ndar)

### Ubuntu Forums

- Transforming your Installation into a Live DVD/CD: <http://ubuntuforums.org/showthread.php?t=688872>
- Proyecto UnionFS: <http://www.filesystems.org/project-unionfs.html>

### Recursos para personalizar el escritorio Gnome

- <http://www.gnome-look.org>
- <http://art.gnome.org/>

### Editor de configuración de Gnome

- <http://library.gnome.org/users/gconf-editor/index.html.es>

El motivo de volver a crearlos inicialmente vacíos es prevenir posibles fallos en el arranque (Listado 6). Y finalmente ya podemos abandonar la jaula: # exit

### Preparación del árbol de directorios del CD-DVD

En este apartado se describen los pasos para ultimar el contenido que formará parte del CD o DVD. En primer lugar, copiaremos las imágenes del kernel, `initramfs` y `mementest` procedentes del entorno `chroot` (jaula) (Listado 7).

El hecho de utilizar el instalador Ubiquity implica la necesidad de crear `filesystem.manifest` y `filesystem.manifest-desktop`.

El propósito de hacer esto ya se explicó en la sección anterior sobre la estructura interna genérica de un Live CD (Listado 8). Ya podemos desmontar `/dev`, `/proc` y `/sys`:

```
sudo umount ${WORK}/rootfs/dev
sudo umount ${WORK}/rootfs/proc
sudo umount ${WORK}/rootfs/sys
```

Ahora, convertiremos el árbol de directorios a una imagen SquashFS.

Hay que tener en cuenta que, a pesar de disminuir el espacio requerido en el soporte de grabación, la compresión `squashfs` también implica una ligera penalización en el rendimiento. Por tanto, consideramos dos opciones:

- Opción 1: Generación `squashfs` con compresión: `sudo mksquashfs ${WORK}/rootfs ${CD}/${FS_DIR}/filesystem.${FORMAT}`
- Opción 2: Generación `squashfs` sin compresión y chequeo de datos (UEX Linux): `sudo mksquashfs ${WORK}/rootfs ${CD}/${FS_DIR}/filesystem.${FORMAT} -info -noI -noD -noF`
- `- check_data`

El tiempo empleado en la creación del sistema de archivos `squashfs` depende de la potencia del equipo. Éste se verá incrementado notablemente si se elige la opción con compresión.

La segunda opción es más recomendable cuando el soporte de grabación es DVD y el tamaño total no excede los 4 GB. En este caso, se puede prescindir de la compresión añadiendo los parámetros `-noI -noD -noF` en el comando `mksquashfs`.

Establecemos GRUB como gestor de arranque del Live CD:

```
sudo find /boot /usr/lib/grub/ -iname 'stage2_eltorito' -exec cp -v {} ${CD}/boot/grub \;
```

Editamos la configuración de GRUB con Gedit o cualquier otro editor de textos:

```
sudo gedit ${CD}/boot/grub/menu.lst
```

El contenido de `menu.lst` ha de asemejarse al que se muestra a continuación para UEX Linux (Listado 9). Por último, se calculan los resúmenes MD5 de todos los ficheros del CD:

```
cd ${CD} && find . -type f -print0 |
xargs -0 sudo md5sum |
sudo tee ${CD}/md5sum.txt
```

### Generar y comprobar la imagen ISO

Ya sólo falta generar la imagen ISO-9660 que grabaremos en CD o DVD según el espacio requerido. En el caso de la distribución UEX Linux alpha 1, éste fue el comando `mkisofs` utilizado en la generación de `uexlinux2008a1-livedvd.iso` (Listado 10).

Para comprobar el correcto funcionamiento de la nueva distribución Linux personalizada hay dos opciones:

- La primera es arrancar la imagen ISO en una máquina virtual (VirtualBox, por ejemplo).
- La segunda alternativa es grabarla en CD o DVD para ejecutarla directamente sobre el equipo físico.

Si algo no ha funcionado tal y como esperábamos tendremos que encontrar la causa,

solucionar el problema y repetir el proceso de remasterización.

### Conclusiones

Crear una distribución adaptada específicamente a las necesidades y propósitos de un grupo de usuarios o una institución es relativamente simple, siempre que se dispongan de conocimientos medios en Linux. Las ventajas que pueden llegar a aportar son muy considerables, ya que se asegura que la distribución está perfectamente ajustada a los requisitos de uso de la institución. Se crea una imagen corporativa, se estandarizan los paquetes que serán usados, se elimina el software innecesario, se incorporan las aplicaciones más eficiente según las necesidades y, además, los usuarios no necesitan (uno a uno) realizar el esfuerzo de instalar los paquetes que utilizan habitualmente en su empresa u organización. En relación al software proporcionado con la distribución, otra opción interesante es la creación de repositorios propios con versiones específicas de aplicaciones y librerías que usarán los usuarios. En este artículo se ha mostrado una guía técnica muy elemental de ejemplo sobre cómo realizar paso a paso una distribución Linux. En cambio, dependiendo del número y tipo de usuarios que la utilizarán, quizás el esfuerzo mayor conlleve la selección de los paquetes que formarán parte de la distribución. No sólo hay que comparar la idoneidad de las diferentes aplicaciones alternativas y su compatibilidad entre sí, sino también poner de acuerdo y convencer a los usuarios que las opciones elegidas son las mejores.



### Sobre los autores

**Adolfo Lozano Tello** es profesor del Departamento de Ingeniería de Sistemas Informáticos y Telemáticos de la Universidad de Extremadura y Director del Centro Internacional de Referencia LINUX. Fue Director del Servicio de Informática de la Universidad de Extremadura desde marzo de 2004 a septiembre de 2005. Es Licenciado en Informática por la Universidad de Granada desde 1993 y Doctor en Ingeniería Informática por la Universidad de Extremadura desde 2002 obteniendo el Premio Extraordinario de Doctorado. Sus líneas de investigación se centran en la Ingeniería de Ontologías y la Web Semántica. Es profesor en el Máster de Software Libre de la Universidad de Extremadura.

**Marcos Blanco Galán** es Ingeniero en Informática e Ingeniero Técnico en Informática de Sistemas. Su labor investigadora se centra en la consultoría, soporte y administración de sistemas basados en Linux y migración a entornos FLOSS. Actualmente trabaja en el Centro Internacional de Referencia Linux (<http://cirl.unex.es>) y es profesor en el Máster de Software Libre de la Universidad de Extremadura. El Centro Internacional de Referencia Linux (CIRL) es una entidad sin ánimo de lucro suportada por la Junta de Extremadura y la Universidad de Extremadura creada para ayudar a potenciar el uso del sistema operativo Linux en empresas e instituciones. Los objetivos principales del CIRL son, por un lado, impulsar el uso de Linux como una alternativa frente a sistemas operativos propietarios y, por otro lado, servir de soporte a particulares, PYMES e instituciones extremeñas que deseen migrar sus sistemas a Linux. (<http://cirl.unex.es>)