

**Desarrollo**  
QTCanvas: un lienzo  
para tus aplicaciones

**Etiqueta**  
Aprende a pronunciar  
"GNU" con Paul C. Brown

**Desktopía**  
Skippy: previsualización  
de ventanas activas

 **KDict**  
Un diccionario  
digital

**Edición en  
Castellano**

LINUX MAGAZINE  
NÚMERO 05

**LINUX**  
MAGAZINE

# LINUX

## MAGAZINE

NÚMERO 5 • P.V.P 4,95 €

**Buzones seguros y libres de SPAM**

# EMAIL SEGURO

Organiza tu correo en tablas HTML con Hypermail p8

Filtrado de mensajes con SpamAssasin p11

Encriptación de correo con GnuPG p14

## Seguridad Wireless II

Utilizamos un Zaurus para hackear una red  
protegida con encriptación WEP p75

## Gimp

Programación de guiones  
en Python p44

## Bluetooth p49

Controla tu sistema Linux  
desde tu móvil



Email Seguro

Filtrado SPAM

Encriptación GnuPG

Seguridad Wireless II

GIMP con Python

Bluetooth

QTCanvas

DEMU

Emulación de Sistemas con QEMU

# MUNDOS VIRTUALES



Figura 1: Knoppix 3.7 arrancado desde la imagen ISO.

¿Has deseado alguna vez poder ejecutar Linux dentro de Linux? QEMU es una aplicación de código abierto que nos permite emular un entorno de hardware completo dentro de nuestro sistema Linux

## POR FABRIZIO CIACCHI

Linux soporta una serie de aplicaciones que nos permiten emular las condiciones de una arquitectura hardware. Estas aplicaciones pueden crear un equipo virtual dentro de nuestra máquina Linux. Podemos usar esta máquina virtual para probar software o incluso ejecutar otro sistema operativo. Una aplicación que emule un entorno de hardware normalmente se conoce como un emulador de sistema. El emulador de sistemas más común para Linux es Bochs [1], un emulador muy potente que es habitualmente muy difícil de configurar, y VMware [2], un muy buen y muy rápido emulador que, desafortunadamente, es caro al ser un producto comercial. Pero otro competidor en el campo de los emuladores de sistemas ha hecho su aparición en escena. En este artículo aprenderemos lo que hay que saber sobre la potente (y gratuita) aplicación de emulación de sistemas QEMU.

QEMU es extremadamente fácil de usar, proporcionando simples comandos para tareas que pueden resultar difíciles con otros emuladores. Os mostraremos como usar QEMU en circunstancias reales, pero debéis recordar que este artículo solo cubre una fracción de las funciones y comandos disponibles en QEMU. Para ver el resto debes descargar QEMU y ponerlo en marcha.

### Instalación de QEMU

QEMU está disponible como código fuente o como binario Linux precompilado. Encontraremos la versión binaria precompilada en la página Web de QEMU [3]. Descargaremos la versión binaria al directorio root ("/). Abriremos una consola, ponemos el archivo en el directorio, nos convertimos en root e introducimos los siguientes comandos:

```
$ cd /
$ su
- Insertar contraseña root ↵
y presionar INTRO-
# tar zxvf
qemu-0.6.1-i386.tar.gz
# qemu (to test)
```

El programa desempaquetará todos los archivos necesarios en el lugar adecuado para el sistema Linux. Si tenemos un problema con la versión precompilada de QEMU o si deseamos instalar QEMU desde el código fuente tendremos que descargarnos la última versión e introducir los siguientes comandos desde una consola:

```
$ su
- Insertar contraseña root ↵
y presionar INTRO-
# tar zxvf qemu-0.6.1.tar.gz
```

```
# cd qemu-0.6.1
# ./configure
# make
# make install
# qemu (to test)
```

Debemos arrancar QEMU desde dentro de una ventana del entorno X Window. Cuando arranquemos QEMU empezará emulando el entorno hardware de nuestro equipo. Si tenemos un Pentium II, QEMU emulará un Pentium II; si estamos en un PowerPC, QEMU emulará un PowerPC. Para emular una arquitectura hardware distinta debemos especificar el nombre de la arquitectura con el comando *qemu*. Para conocer la lista de arquitecturas soportadas por QEMU introduciremos *qemu-* y presionamos la tecla TABULADOR dos veces.

## Inicio de un Live CD

Podríamos usar QEMU para probar una imagen ISO que hayamos descargado de Internet. Por ejemplo, si encontramos una imagen ISO de un Live CD de una distribución Linux en el DVD de Linux Magazine, podremos usar QEMU para probar la distribución. Por ejemplo, suponiendo que tenemos una imagen ISO de KNOPPIX [4] llamada *knoppix.iso*, si queremos probar la imagen antes de masterizarla, podemos abrir una consola, hacernos root (para asegurarnos de que podemos acceder a los periféricos sin problemas) e introducir este comando:

```
$ su
- Insertar contraseña root ↵
y presionamos INTRO -
# qemu -cdrom knoppix.iso
```

Se abrirá otra ventana y comenzará la emulación como si el programa se estuviera ejecutando desde hardware real. Knoppix presenta un arranque gráfico y, tras la elección del método de arranque, comienza en modo gráfico. Podemos arrancar y usar KNOPPIX como hacemos siempre, con las limitaciones obvias de un emulador, que probablemente sea más lento que un sistema ejecutándose directamente en el hardware.

El modo más fácil de usar Internet y comunicarse con el entorno del anfitrión es usar la opción *-user-net*:

```
# qemu -user-net -cdrom ↵
```



Figura 2: Gentoo, instalado en hda2, ejecutándose bajo Debian, instalado en hda1.

```
knoppix.iso
```

Si hay un servidor Samba instalado en el sistema, el entorno emulado puede acceder al entorno del anfitrión (via Samba) con la opción *-smb <directory >*. Esta opción para acceder SMB solo puede utilizarse con la opción *-user-net*.

Si tenemos un CD que contenga una distribución Linux o un Live CD Linux, podemos ejecutarlos con un simple comando:

```
# qemu -user-net -cdrom ↵
/dev/cdrom
```

Con esta opción insertamos el CD en el dispositivo de CDs sin montarlo. QEMU usa el archivo del dispositivo en lugar de la imagen ISO para iniciar la emulación. Esta es una opción muy útil que hace que dispositivos como CDs o discos flexibles sean accesibles tanto al sistema anfitrión como al sistema emulado.

## Uso del Mismo Disco Duro

Otra situación en la cual podemos usar la ruta del dispositivo en lugar del nombre de un archivo imagen es la representada por el comando QEMU que permite arrancar un sistema operativo en un disco duro. Un ejemplo típico es un disco duro con dos versiones diferentes de Linux, por ejemplo Debian y Gentoo.

¿Qué ocurre si estamos en Debian y queremos arrancar Gentoo? En situaciones normales, tenemos que cerrar todos los programas, desconectarnos del entorno X y reiniciar el sistema. Con QEMU podemos arrancar un sistema operativo en nuestro disco duro sin necesidad de reiniciar:

```
# qemu -snapshot -hda /dev/hda
```

La opción *-snapshot* especifica que todas las modificaciones realizadas serán escritas en un archivo temporal del disco en lugar de en el propio disco. Esta opción ayuda a prevenir la pérdida de datos en situaciones en las que la emulación se realiza en el mismo disco donde se ubica el sistema anfitrión. Si tenemos un cargador de arranque como GRUB instalado en MBR, veremos el arranque del sistema emulado. Una vez se inicia el sistema lo podemos usar normalmente (figura 2).

## RAM Virtual

La opción de QEMU *-m* nos permite especificar la cantidad de memoria RAM virtual (en MB) para usar en la emulación (el valor por defecto es 128 MB). Si tenemos mucha memoria RAM física especificaremos más memoria RAM virtual para incrementar las prestaciones de la emulación. Volviendo al ejemplo anterior, si tenemos 512 MB de memoria

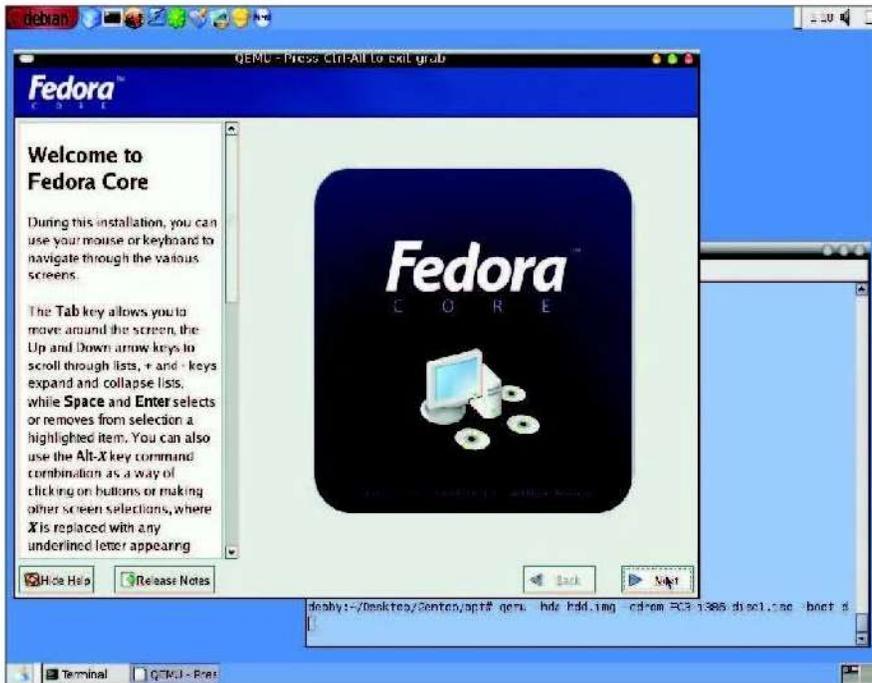


Figura 3: Con una imagen para el disco duro, podemos iniciar la instalación de Linux.

física y queremos asegurarnos las prestaciones adecuadas para emular un sistema Linux deberemos usar el siguiente comando:

```
# qemu -snapshot -m 256
-hda /dev/hda
```

### Instalación de Linux

Si queremos instalar una distribución Linux en el entorno emulado necesitamos crear un archivo que QEMU pueda usar como disco duro virtual. Para crear este archivo podemos usar un programa llamado *qemu-img*, el cual esta disponible con la aplicación QEMU. La sintaxis para usar *qemu-img* es muy simple, pasando el nombre *hdd.img* y el tamaño (en MB) de la imagen que deseamos crear:

```
# qemu-img create
hdd.img 2000M
```

Una vez hemos creado el archivo que servirá de disco duro virtual podemos instalar una distribución Linux directamente desde una imagen ISO. Por ejemplo, podemos descargar la distribución Ubuntu [5] e instalarla en el entorno emulado. Cuando hemos acabado de descargar la imagen ISO de Ubuntu, debemos indicar a *qemu* el archivo que debe usar como disco duro virtual *-hda hdd.img*, la ruta de la imagen ISO a

usarse como CD-ROM y el dispositivo de arranque *-boot d*. QEMU supone por omisión que el arranque se inicia desde el disco duro (si éste está presente):

```
# qemu -hda hdd.img -cdrom
ubuntu.iso -boot d
```

Pero, ¿qué ocurre si queremos instalar una distribución que tiene más de un CD? En este caso necesitamos pasar la opción *-monitor stdio*, por lo que cuando

QEMU lance la emulación se abrirá un interfaz de comandos interactivo en el terminal:

```
# qemu -monitor stdio
-hda hdd.img -cdrom
fedora_cd1.iso -boot d
```

En este interfaz de comandos podemos introducir comandos para gestionar la emulación, los distintos comandos de la interfaz nos permiten reiniciar la emulación, salvar el estado de la emulación para reiniciarla más tarde o cambiar el archivo de un dispositivo particular emulado. Si tenemos imágenes ISO múltiples representando una colección de CDs de instalación de una distribución Linux podemos cambiar al segundo CD del conjunto de instalación con un comando como el siguiente:

```
# qemu change cdrom
fedora_cd2.iso
```

Cuando hemos acabado el proceso de instalación tenemos una imagen del disco duro que podemos arrancar con QEMU. Para iniciar la emulación escribimos el siguiente comando:

```
# qemu hdd.img
```

En este caso no usamos ninguna otra opción para el ejecutable porque el parámetro por defecto pasado a QEMU

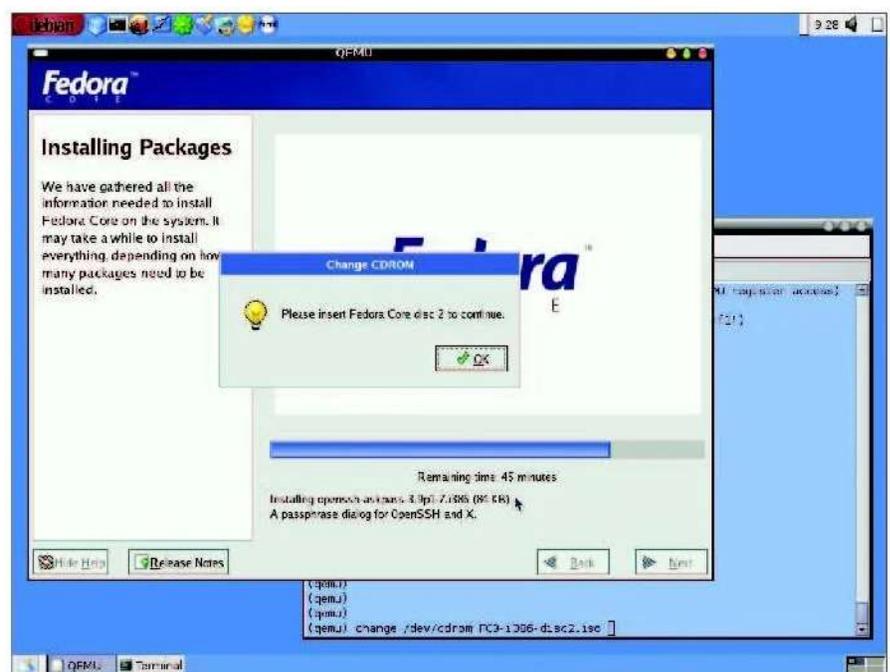


Figura 4: Con el monitor habilitado podemos cargar la imagen ISO de un dispositivo.



Figura 5: Una sesión SEAL bajo QEMU.

tiene como objeto el disco duro principal *hda*.

### ¿Por qué no DOS?

¿Quién no se acuerda del DOS? Un escenario que una vez fue común (y aún puede serlo) en el caso de que nuestra organización tenga importantes programas que sólo corran en DOS. En lugar de crear una partición de 50 MB para el programa DOS y salir de Linux cada vez que lo necesitamos, podemos usar QEMU con una imagen previamente hecha de FreeDOS [6] (un DOS distribuido con la licencia GPL). Podemos descargar FreeDOS desde [7].

Cuando hemos descargado una imagen comprimida de FreeDOS, *fdos-100meg.tar.gz*, la abrimos con *fileroller* o con *ark* y extraemos el archivo *fdos\_8h1.img* en un directorio. Como root ejecutamos este comando:

```
# qemu -hda fdos_8h1.img -fda /dev/fd0 -boot c
```

Debemos saber que hemos pasado la opción *-fda* a QEMU. Al igual que las opciones *-hda* y *-cdrom*, esta opción se utiliza para leer el contenido del disco blando en el entorno emulado. FreeDOS entonces se inicia y está completamente operativo. Podemos, por tanto, crear un directorio para nuestro programa DOS y

copiar los archivos que necesitamos en el. En el FreeDOS emulado escribiremos:

```
C:\> mkdir program
C:\> copy A:\*. * C:\program
```

### SEAL

Ahora que tenemos un DOS funcionando lo podemos usar para otras cosas. Por ejemplo podemos usar el programa SEAL [8]. SEAL es un entorno gráfico para DOS (como Windows 3.1) que viene con algunas funciones similares a las proporcionadas por sistemas más avanzados como Windows 98. Los archivos de instalación de SEAL están en el directorio *C:\fdos\seal2*. Algunas veces el programa no se inicia debido a problemas en la gestión de la memoria, por lo que también debemos usar las utilidades que vienen para la gestión de intercambio:

```
C:\> cd C:\fdos\seal2
C:\> cwsparm
C:\> cwsdpmi
C:\> cwsdpr0
C:\> install
```

Una vez que SEAL está instalado, necesitamos reiniciar los programas de intercambio e iniciar el programa del ratón, y entonces iniciar SEAL (figura 5).

```
C:\> cwsdpmi
C:\> cwsdpr0
C:\> cd C:\seal2
C:\> ctmouse
C:\> seal
```

### Conclusión

QEMU es una aplicación de emulación muy potente. Como otros emuladores, QEMU sufre problemas de rendimiento. Una aplicación no puede correr tan rápido dentro de QEMU como lo haría en el sistema anfitrión. Por otro lado, la velocidad de QEMU es comparable a la de otros emuladores como Bochs. Podemos usar QEMU para comprobar otros sistemas operativos como NetBSD o BeOS.

Encontraremos un gran archivo de imágenes de sistemas operativos en FreeOSZoo [9], pudiendo también usar imágenes construidas desde Bochs [10]. Algunos sistemas operativos pueden incluso tener imágenes de QEMU (o Bochs) que podemos descargar.

### RECURSOS

- [1] Página Web de Bochs: <http://bochs.sourceforge.net>
- [2] Página Web de VMware: <http://www.vmware.com>
- [3] Página Web de QEMU: <http://fabrice.bellard.free.fr/qemu>
- [4] Página Web de KNOPPIX: <http://www.knoppix.net>
- [5] Página Web de Ubuntu: <http://www.ubuntulinux.org>
- [6] Página Web de FreeDOS: <http://www.freedos.org>
- [7] Imagen Bochs FreeDOS de 100 MB: <http://prdownloads.sourceforge.net/bochs/fdos-100meg.tar.gz?download>
- [8] Página Web de SEAL: <http://sealsystem.sourceforge.net>
- [9] Página Web de FreeOSZoo: <http://www.freeoszoo.org>
- [10] imágenes Bochs: [http://sourceforge.net/project/showfiles.php?group\\_id=12580&package\\_id=27799](http://sourceforge.net/project/showfiles.php?group_id=12580&package_id=27799)

### EL AUTOR

Fabrizio Ciacchi (<http://fabrizio.ciacchi.it>) es un estudiante Italiano de Ciencias Informáticas en la Universidad de Pisa. También trabaja como consultor y escribe artículos sobre Linux.