



Juegos en GNU/Linux:
Doom y derivados
(Usando el puerto Zdoom)

Eye del Cul

1. Introducción

Hace tiempo, poco después de comenzar en [Arcades 3D](#) con esta sección de “Juegos en Linux”, **Karnizero** nos explicó cómo instalar el puerto Doom Legacy —y quien esté interesado puede [consultar el tutorial correspondiente](#)—, con el que podíamos jugar cómodamente a *Doom* o *Heretic*... Y bueno, ahora me toca a mí hacer lo propio con el puerto **Zdoom**.

Zdoom es, al igual que Doom Legacy, un puerto que permite ejecutar los viejos títulos basados en el «*Doom engine*» en nuestros sistemas modernos: en este caso, Linux y las variantes recientes de Windows, aunque con un poco de ingenio también es posible hacerlo funcionar en FreeBSD... Esto es importante porque, tratándose de juegos que aparecieron en la primera mitad de la década de los noventa —hace más de diez años—, sus versiones originales sólo funcionaban en MS-DOS.

¿Y por qué Zdoom? Bueno, hay gente a la que no le acaba de gustar Doom Legacy... Y además, Zdoom tiene algunas ventajas destacables. Por ejemplo, permite arrancar más juegos —es totalmente compatible con *Hexen* y *Strife*, y las últimas versiones incluyen también compatibilidad con *Chex Quest*—. Y el sonido resulta más fácil de configurar; puede emular las melodías midi de los juegos originales con el sintetizador que trae incorporado, o bien mediante Timidity. Para que nos entendamos: ¡con Zdoom hay música! —con Doom Legacy se supone que también..., para quien sepa cómo configurarla—.

2. Instalando Zdoom en GNU/Linux

2.1. Requisitos

Así que vamos a ponernos manos a la obra. Para ejecutar Zdoom necesitamos lo siguiente:

- Librerías SDL:
 - Con la versión 2.7.1 y las anteriores, usad la rama tradicional de SDL (la 1.2).
 - A partir de la 2.8.0 necesitaréis las más modernas SDL2.
- Librería de sonido Fmod:

- Para versiones de Zdoom anteriores a la 2.3.0, usad la versión 3.75.¹
 - De la versión 2.3.0 de Zdoom en adelante, se requieren las versiones 4.x de esta librería (por ejemplo, mientras escribo esta frase veo que la más reciente corresponde con el fichero `fmodapi44429linux.tar.gz`).
- Versiones antiguas de Zdoom requieren las librerías libFLAC y libFLAC++.
 - Un entorno gráfico (el X Window System).
 - Timidity (recomendable aunque no imprescindible).
 - Si vais a usar la versión 2.3.0 de Zdoom o más recientes, os vendrá bien disponer de las librerías GTK2. La copia que os ofrecemos aquí, en *Arcades 3D*,² fue compilada sin ellas y no las necesita, pero si descargáis por ahí otros binarios distintos quizás sí os hagan falta.
 - El archivo `.WAD` del juego que queramos ejecutar (*Doom*, *Doom 2*, *Heretic*...)

Probablemente casi todo esto vendrá ya incluido en la mayoría de las distribuciones grandes —Debian, Ubuntu, Mandriva, etc—. Quizás Timidity —una utilidad para reproducir melodías midi— y la librería Fmod sean lo más problemático de instalar, puesto que no estarán disponibles en muchos repositorios; así que he escrito un anexo al final de este tutorial, con las instrucciones necesarias para tener ambas cosas instaladas y funcionando.

2.2. Configuración e Instalación

Como ya hemos visto en otros manuales de esta serie, podemos descargarnos unos binarios listos para usar, o conseguir el código fuente del Zdoom para compilarnos unos propios. Obviamente, lo primero es lo más sencillo, y lo que explicaré a continuación.

Daré por supuesto que ya tenéis Timidity y la librería Fmod bien instalados y configurados —consultad el anexo, a partir de la página 14, si tenéis dudas sobre ello—. En realidad, Timidity no es imprescindible para poder disfrutar de Zdoom, ni siquiera para la música. Zdoom puede reproducir directamente las melodías MUS —el formato nativo de *Doom*, *Heretic* y demás,

¹Disponible en <http://www.arcades3d.net/linux/archivos/fmodapi375linux.tar.bz2>.

²<http://www.arcades3d.net/linux/archivos/zdoom-2.8.1-x86.tar.xz>.

una variación de los midis que encontramos habitualmente—; sin embargo, algunos mapas creados por usuarios sí emplean música en el formato midi original, y si queremos oírla necesitaremos Timidity.

Seamos ordenados; antes que nada, crearemos una carpeta en la que colocar nuestros juegos:

```
$ pwd
/home/usuario
$ mkdir juegos
$ cd juegos
```

Ahora tendremos que descargar los binarios del Zdoom. En la sección «Linux» de *Arcades 3D* podéis encontrarlos. Os ofrecemos varias versiones:

- Versión 2.1.7 ([zdoom-2.1.7_bin.tar.bz2](#))
- Versión 2.8.1 ([zdoom-2.8.1-x86.tar.xz](#))

Descargad la que más os convenga (la versión 2.1.7 fue publicada en 2006 así que tiene ya su tiempo, pero correrá mejor en ordenadores viejos). Podéis usar Wget desde una terminal:

```
$ wget http://www.arcades3d.net/linux/archivos/zdoom-2.8.1-x86
.tar.xz
```

Estos binarios deberían funcionar en la mayoría de los sistemas. Los compilé en un Pentium IV, con Glibc 2.5 y GCC 4.2.4, y he procurado reducir las dependencias al mínimo para facilitar la compatibilidad. Como muestra, en el ejemplo de arriba (`zdoom-2.8.1-x86.tar.xz`) serían las siguientes:

```
$ readelf -d zdoom
(NEEDED)                Shared library: [libSDL2-2.0.so.0]
(NEEDED)                Shared library: [libpthread.so.0]
(NEEDED)                Shared library: [libfmodex-4.22.02.so]
(NEEDED)                Shared library: [librt.so.1]
(NEEDED)                Shared library: [libsndfile.so.1]
(NEEDED)                Shared library: [libmpg123.so.0]
(NEEDED)                Shared library: [libdl.so.2]
(NEEDED)                Shared library: [libstdc++.so.6]
(NEEDED)                Shared library: [libm.so.6]
(NEEDED)                Shared library: [libgcc_s.so.1]
(NEEDED)                Shared library: [libc.so.6]
[...]
```

Así que acabamos de descargarnos el Zdoom, comprimido en un archivo `.tar.bz2`; lógicamente, hay que descomprimirlo:

```
$ xzcat zdoom-2.8.1-x86.tar.xz | tar -xvf -
```

Esto crea un directorio nuevo dentro del actual, llamado en este caso `zdoom-2.8.1-x86` —no digáis que no lo veáis venir, ¿eh?—. En él os esperan los tres ficheros que requiere Zdoom para funcionar: el ejecutable `zdoom`, uno de datos con el nombre `zdoom.pk3`, y por último una pequeña librería, `liboutput_sdl.so`.

Y ahora sólo nos falta el archivo WAD del juego que nos interese ejecutar con Zdoom. Por ejemplo, en el caso de *Doom 2* sería algo así:

```
$ cp ~/juegos/doom2/doom2.wad ~/juegos/zdoom
```

...Vamos, esto suponiendo que tenemos una copia del *Doom 2* instalada en la carpeta `/home/usuario/juegos/doom2`. Zdoom puede funcionar con varios juegos: *Doom*, *Doom 2*, *The Ultimate Doom*, *Heretic*, *Hexen*, *Strife*..., y quizás algún otro que no recuerde ahora 🤔.

En fin, hecho esto ya podríamos arrancar el Zdoom. Bastaría con acceder a la carpeta en la que lo hemos instalado, y escribir:

```
$ ./zdoom
```

La primera vez que ejecutemos Zdoom se creará automáticamente una carpeta oculta en el directorio raíz de nuestro usuario —como hacen muchos otros programas—, situada bien en `~/zdoom`, bien en `~/config/zdoom`.³ En ella se almacenarán nuestras partidas guardadas, así como el archivo de configuración `zdoom.ini`. Éste es importante, por cierto.

Para escuchar las melodías del juego, arrancaremos Zdoom del modo habitual. Tendremos que acceder, en el menú principal, a «*Options* → *Sound Options* → *Advanced Options*»; y una vez aquí, activamos la opción «*Use FM synth to MUS Music*». Esto hace que la música en formato MUS —el formato nativo de *Doom*, similar al midi— se reproduzca mediante sonido digitalizado. A partir de la versión 2.3.0, la opción correspondiente se encuentra en «*Options* → *Sound Options* → *Midi Device*».

³El lugar exacto depende de la versión.

No obstante, y como ya comenté al principio, Zdoom también puede reproducir midis corrientes. De hecho, hay algunos mapas circulando por Internet que emplean melodías en formato MID, en lugar del formato MUS tradicional de *Doom*. Si deseamos oír la música en estos casos, deberemos tener bien instalado Timidity; y es posible que haya que modificar el archivo de configuración de Zdoom⁴ del siguiente modo:

```
timidity_extargs=-Os
```

De este modo especificamos que Timidity reproduzca la música mediante ALSA —para lo cual necesitaremos haberlo compilado con soporte para ALSA; en el anexo lo explico con detalle—. Puede ser necesario cambiar esta línea, o no; depende de cómo tengamos configurado el sonido en nuestro ordenador.

2.3. ¡Zdoom se cuelga!

A algunas personas —yo entre ellas 🤔— les ha sucedido esto la primera vez que han intentado comenzar una partida. Una forma de solucionarlo es editar el archivo de configuración del Zdoom, que suele estar colocado en `~/zdoom/zdoom.ini`. En este fichero encontraremos probablemente una línea como ésta:

```
snd_output=default
```

Pues bueno, bastaría con sustituir «*default*» por «*alsa*» —o por «*oss*», si esto no da resultado, o no escucháis ningún sonido al jugar con Zdoom—:

```
snd_output=alsa
```

Y por último, si este archivo (`zdoom.ini`) no existe, basta con cargar Zdoom y salir inmediatamente de él —con la opción de «*Quit Game*», sin comenzar ninguna partida— para que sea creado automáticamente.

⁴Como ya se ha dicho, `~/zdoom/zdoom.ini` o `~/config/zdoom/zdoom.ini`.

3. Capturas de Pantalla

Bueno, no pueden faltar algunas capturas de pantalla que demuestren lo comentado en este tutorial:



En esta primera imagen estoy jugando el primer nivel de *Doom 2*, usando la versión 2.1.0 de Zdoom. El ordenador es un Pentium II a 350 MHz, funcionando con Debian GNU/Linux y un bonito escritorio Fluxbox. Tomada en 2006.

Como no sólo de *Doom* vive el hombre, en esta otra captura de pantalla podéis ver cómo se ejecuta *Strife* con la versión 2.3.0 de Zdoom. La compatibilidad es completa, y el juego puede disfrutarse sin ningún inconveniente.



Esta imagen se tomó en febrero de 2009 para hacer un poquito más variado este manual —la anterior *también* trataba de *Doom*—, en mi “nuevo” Pentium III.

Os dejo una última imagen, para hacer este tutorial un poco más ameno:



Aquí estoy jugando una modificación del *Doom 2* llamada «*Pinochestein 3D*», que trae un buen surtido de gráficos y sonidos propios, y que como se puede comprobar funciona perfectamente en Zdoom bajo Linux. Graciosa, a su manera.

4. Compilando Zdoom

A la gente que no quiera complicarse mucho la vida probablemente le interesará más usar los binarios ya compilados, tal y como acabo de explicar en la primera parte de este manual. Sin embargo, puede haber quien necesite compilarse unos propios porque los anteriores no le funcionen, o porque quiera tener Zdoom lo más optimizado posible para su máquina, o simplemente porque le guste hacer las cosas así. Si éste es tu caso, sigue leyendo...

Hasta la llegada de la versión 2.1.0, compilar Zdoom en Linux podía resultar un auténtico dolor de cabeza. Había que aplicar un parche al código fuente que hubiésemos descargado —parche que estaba escondido en un hilo del foro oficial de Zdoom—, usar unas versiones muy concretas de NASM y la librería Fmod, y cruzar los dedos para que todo funcionase sin problemas... Afortunadamente ya no es así; y la versión 2.1.0, que al parecer es la siguiente versión *oficial* desde la ya lejana 1.22, puede compilarse sin mayores complicaciones.

4.1. Versiones 2.1.x

Bueno, he mencionado la versión 2.1.0, pero lo comentado aquí también sirve para la 2.1.1 —aparecida mientras preparaba este manual— y las siguientes. Sin embargo, a partir de la versión 2.3.0 los autores de Zdoom han hecho varios cambios importantes, y por eso el modo de compilarlas se explica más adelante (mirad en la sección 4.2).

4.1.1. Requisitos

Y basta de charla..., pongámonos ya manos a la obra. Para compilar Zdoom vamos a necesitar lo siguiente:

- [Código fuente de Zdoom](#) (disponible aquí mismo, en *Arcades 3D*, o en la [página oficial](#)).
- Librerías SDL.
- Librería de sonido Fmod, en su versión 3.75.
- Ensamblador NASM.
- libFLAC y libFLAC++.

Si no tenéis alguna de estas librerías, probablemente podréis conseguirla en los repositorios de vuestras distribuciones de Linux. Y si no es así, siempre se pueden encontrar en sus páginas oficiales. Con la excepción, quizás, de Fmod: como la versión 3.75 tiene ya bastantes años a sus espaldas, sus autores han dejado de distribuirla, y resulta un tanto difícil de localizar. Sin embargo, en *Arcades 3D* pensamos mucho en vosotros, así que hemos subido una copia:

- SDL: <http://www.libsdl.org>.
- Fmod: <http://www.arcades3d.net/linux/archivos/fmodapi375linux.tar.bz2>.
- NASM: <http://sourceforge.net/projects/nasm>.
- libFLAC y libFLAC++: <https://xiph.org/flac/>.

De todas ellas, quizás la más problemática de instalar sea Fmod; consultad el anexo si tenéis alguna duda con esto... De momento asumiré que ya está instalada en vuestras máquinas.

4.1.2. Compilación e Instalación

Bueno, empezaremos por descargarnos el código fuente del Zdoom, y crear una carpeta para guardarlo y descomprimirlo:

```
$ mkdir zdoomsrc
$ cd zdoomsrc
$ wget http://www.arcades3d.net/linux/archivos/zdoom-2.1.7_src.tar.bz2
```

Y ahora extraemos el contenido del archivo `tar.bz2`:

```
$ tar -jxvf zdoom-2.1.7_src.tar.bz2
```

Esto creará una nueva carpeta llamada `zdoom`, dentro de la que acabamos de crear (`zdoomsrc` en este ejemplo). En ella encontraremos todo el código fuente que necesitamos compilar.

Sin embargo, hay que pulir algún detallito antes de ponernos a la tarea. Por algún motivo el compilador busca algunas de las dependencias —las librerías necesarias que ya he mencionado— en el directorio `/usr/lib`, y no en `/usr/local/lib` como suele ser habitual en algunas distros de Linux. No

sé si sucederá siempre así, pero es un problema que he encontrado en las dos máquinas en que lo he probado —una con Debian GNU/Linux, y la otra con Vector Linux—.

Algo muy fácil de solucionar, por otra parte. Basta con crear algunos enlaces simbólicos en `/usr/lib` y `/usr/include` que apunten a las librerías originales. Para ello probablemente tendréis que acceder a la cuenta del administrador:

```
$ su
Password:
# ln -s /usr/local/fmod/lib/libfmod-3.75.so /usr/lib/libfmod.so
# ln -s /usr/local/fmod/include/fmod /usr/include/fmod
# ln -s /usr/local/fmod/include/fmod/fmod.h /usr/include/fmod.h
# ln -s /usr/local/lib/libFLAC.so /usr/lib/libFLAC.so
# ln -s /usr/local/lib/libFLAC++.so /usr/lib/libFLAC++.so
# exit
$
```

Una vez resuelto este pequeño inconveniente —si es necesario hacerlo— ya podemos compilar el Zdoom. Para ello accederemos a la carpeta llamada `zdoom`, y con una sencilla orden `make` todo echará a andar:

```
$ pwd
/home/usuario/zdoomsrc
$ cd zdoom
$ make
```

Puede costarle un buen rato terminar, sobre todo en las máquinas más viejas. En mi Pentium II a 350 MHz la compilación se prolongó durante veinte minutos más o menos... Lo digo para que no os desesperéis si tarda más de lo que esperabais.

Cuando termine, habrán aparecido los siguientes archivos:

- `zdoom` (ejecutable).
- `zdoom.map`.
- `wadsrc/zdoom.pk3`.
- `zdoom.pk3` (enlace simbólico al anterior).

Y con esto hemos terminado... Sólo nos faltaría copiar el archivo WAD del juego que nos interese —`doom2.wad` para jugar a *Doom 2*, `heretic.wad` para el *Heretic*, etc— a esta misma carpeta; o, más recomendable, hacer lo contrario (copiar los archivos listados arriba en el directorio del juego).

Nada más. Que lo disfrutéis...

4.2. Versiones 2.3.x y más recientes

El modo de compilar estas versiones de Zdoom cambia respecto a las anteriores, y por eso he decidido dedicarles una sección por separado. Como hicimos antes, veremos primero los requisitos necesarios.

4.2.1. Requisitos

Bien, vamos a necesitar lo siguiente:

- Una versión moderna de GCC. Aunque en la wiki de Zdoom mencionaban la 3.4.6, yo la he probado y no funciona. Mi recomendación es que uséis alguna de la serie 4.2.x, o más recientes.
- CMake 2.6 o 2.4.
- Librerías SDL .
- Librería de sonido [Fmod](#); usad la versión 4.20 y posteriores.
- Ensamblador [NASM](#).
- libFLAC y libFLAC++ (quizás, depende de la versión; con la 2.8.1, por ejemplo, no hacen falta).
- GTK2 (opcional).

Notad que GTK2 es opcional. Si tenéis instaladas estas librerías, Zdoom las usará, y de lo contrario se compilará sin ellas.

Respecto a lo demás, lo recomendable es que uséis los gestores de paquetes de vuestras distribuciones. Quizás la librería Fmod no se halle en ellos; en tal caso, tendréis que descargarla de su [página oficial](#) e instalarla manualmente en vuestro sistema. No es difícil de hacer; en el anexo se explica cómo.

4.2.2. Compilación e Instalación

El primer paso es descargarse el código fuente de Zdoom y descomprimirlo en una carpeta a vuestra elección. Los autores lo distribuyen en un archivo de tipo `.7z`, en *Arcades 3D* lo hemos empaquetado en un fichero `.tar.xz`; en este último caso, para descomprimirlo requeriréis las XZ Utils,⁵ aunque en la mayoría de sistemas modernos vendrán instaladas ya, y supongo que tampoco faltarán en ningún repositorio.

Por ejemplo, para descargar el código fuente de *Arcades 3D* y descomprimirlo:

```
$ wget http://www.arcades3d.net/linux/archivos/zdoom-2.8.1_src
.tar.xz
$ xzcat zdoom-2.8.1_src.tar.xz | tar -xvf -
```

A continuación —y aquí comienzan las diferencias respecto a las versiones viejas de Zdoom— hay que crear un directorio de trabajo dentro de la carpeta principal del código fuente. Más o menos así (en este ejemplo lo he llamado `compilacion`, pero cualquier otro nombre serviría):

```
$ pwd
/home/usuario/zdoomsrc
$ cd zdoom-2.8.1
$ mkdir compilacion
$ cd compilacion
```

Y ahora usamos CMake de la siguiente manera:

```
$ cmake -DCMAKE_BUILD_TYPE=Release ..
```

¡Fijaos en los dos puntos del final de la línea, son importantes! La gente que tenga experiencia con estos asuntos puede probar en su lugar `ccmake ..`, y ajustar las opciones de compilación a su gusto en la interfaz de texto que aparecerá.

Si CMake ha hecho correctamente su parte y no han surgido problemas, ya estáis listos para comenzar el trabajo. Sin salir del directorio `compilacion`, escribimos:

⁵Su página oficial es <http://tukaani.org/xz/>.

```
$ make
```

Y ya sólo queda esperar. La compilación puede prolongarse bastante rato, y en las máquinas más viejas no sería extraño que superase la media hora. Cuando termine, encontraréis los tres archivos necesarios para ejecutar Zdoom:

- `zdoom` (ejecutable).
- `zdoom.pk3`.
- `liboutput_sdl.so`.

Simplemente copiadlos a la carpeta en la que tengáis instalado el juego que os interese: *Doom*, *Doom 2*, *Heretic*, *Hexen*, *Strife*, y creo que incluso *Chez Quest*.

5. Anexo: Fmod y Timidity

Algunas personas pueden tener problemas a la hora de instalar alguna de estas dos cosas. Por ello he decidido escribir esta última parte del manual, que debería servir de ayuda para quien se sienta perdido.

5.1. Librería Fmod

5.1.1. Versión 3.75

Comencemos con Fmod... Este texto en realidad ya lo he escrito en otros manuales —los de *Duke Nukem 3D* y *Shadow Warrior*—, pero no está de más repetirlo aquí.

En primer lugar nos descargamos el archivo correspondiente —en el momento de escribir esto, `fmodapi375linux.tar.bz2`—, y lo descomprimimos:

```
$ wget http://www.arcades3d.net/linux/archivos/fmodapi375linux.tar.bz2
$ tar -jxvf fmodapi375linux.tar.gz
```

A continuación tendremos que crear varias carpetas dentro de `/usr/local` —para hacerlo es probable que necesitéis acceder a la cuenta del administrador, o usar el comando `sudo`—. Así, por ejemplo:

```
$ su
Password: ← Introducimos la contraseña del administrador...
# mkdir -p /usr/local/fmod/include/fmod
# mkdir -p /usr/local/fmod/lib
```

Y accedemos a los contenidos que acabamos de descomprimir para comenzar a copiar los archivos necesarios. Lo único que nos interesa aquí es la carpeta llamada `api`:

```
# cd fmodapi375linux
# cd api
# cp -v libfmod-3.75.so /usr/local/fmod/lib
# cp -v inc/* /usr/local/fmod/include/fmod
```

Ahora necesitamos crear algunos enlaces simbólicos, para que los diversos programas puedan encontrar fácilmente la librería Fmod cuando lo requieran:

```
# ln -s /usr/local/fmod/lib/libfmod-3.75.so /usr/local/fmod/lib/libfmod.so
# ln -s /usr/local/fmod/lib/libfmod-3.75.so /usr/local/lib/libfmod.so
# ln -s /usr/local/fmod/include/fmod /usr/local/include/fmod
```

Y ya lo tenemos... De todas formas, si pensáis compilar Zdoom vosotros mismos, podéis encontraros con que durante el proceso el ordenador busque la librería Fmod en la carpeta `/usr/lib`, en vez de en `/usr/local/lib`. Para solucionar esto, añadiremos algunos enlaces simbólicos más:

```
# ln -s /usr/local/fmod/lib/libfmod-3.75.so /usr/lib/libfmod.so
# ln -s /usr/local/fmod/include/fmod /usr/include/fmod
# ln -s /usr/local/fmod/include/fmod/fmod.h /usr/include/fmod.h
```

5.1.2. Versiones 4.x

La instalación de estas versiones de Fmod resulta más sencilla que la de las anteriores. Basta con descargarlas, descomprimir el fichero correspondiente, y ejecutar un `make install` desde la cuenta del administrador. Veámoslo con detalle.

Descargar Fmod es tan fácil como acudir a la [página oficial](#) y conseguir la última versión. O bien, usar Wget desde la línea de comandos:


```
$ wget http://www.fmod.org/download/fmodex/api/Linux/fmodapi42636linux.tar.gz
```

En este ejemplo se trata de la versión 4.26, pero para cuando leáis esto pueden haber otras. De hecho, existen también versiones de Fmod para sistemas de 64 bits.

Ahora descomprimos el archivo que hemos descargado:

```
$ tar -zxvf fmodapi42636linux.tar.gz
```

Y hecho esto, procedemos a la instalación. Para ello habrá que usar la cuenta del administrador (o el comando `sudo`, en Ubuntu y otras distribuciones similares):

```
$ cd fmodapi42636linux
$ su
Password:
# make install
```

Y ya está hecho. Como veis, no tenía mucho misterio.

5.2. Timidity

Se trata de una utilidad que nos permite reproducir melodías midi en Linux y otros sistemas. Como ya he explicado anteriormente, no resulta imprescindible para jugar con Zdoom, pero sí puede venir bien para escuchar la música de algunos mapas que no emplean el formato MUS nativo de *Doom*. Aquí explicaré los pasos a seguir para instalarlo y configurarlo... De todas formas, en *Ubuntu-es.org* hay un tutorial muy útil y recomendable sobre esto.⁶

Lógicamente, el primer paso sería descargarnos alguna versión de Timidity. La más reciente a día de hoy creo que es la 2.13; podéis echar un vistazo en su página oficial —<http://timidity.sourceforge.net>— para conseguirla.

⁶<http://www.ubuntu-es.org/node/3777>.

5.2.1. Compilar Timidity

En principio sería preferible descargar paquetes ya compilados para vuestras distribuciones. Pero si por alguna razón deseáis compilar Timidity vosotros mismos —o vuestra distribución no cuenta con dichos paquetes—, el proceso es sencillo. Una vez descomprimido el código fuente, basta con teclear lo siguiente en una consola de comandos:

```
$ ./configure --enable-audio=oss,alsa --enable-server --enable-alsaseq
$ make
$ su
Password: ← Entramos en la cuenta del administrador.
# make install
```

Como podéis ver, la primera orden (`./configure`) admite diversas opciones; son importantes, porque con ellas indicaremos qué características del programa queremos activar. Así, en este ejemplo configuramos Timidity para que emplee OSS o ALSA para reproducir los midis (`--enable-audio=oss,alsa`), cosa que funcionará en la mayoría de sistemas Linux; las otras dos opciones no sabría explicar bien para qué sirven exactamente, pero creo que es recomendable usarlas... 🤔

En fin, si deseáis ver con tranquilidad todas las opciones de configuración que pueden emplearse con Timidity, teclead una orden como ésta:

```
$ ./configure --help | less
```

Aunque de todas formas, Timidity, compilado tal y como muestra el ejemplo anterior, probablemente correrá sin problemas en casi todas las máquinas con Linux y una tarjeta de sonido bien configurada.

5.2.2. Instalando el archivo de sonidos

Bueno, ya tenemos Timidity compilado e instalado en nuestro ordenador... Pero aún no hemos terminado. Lo siguiente que nos hará falta será un archivo de sonidos digitalizados, correspondientes a las notas e instrumentos que emulará Timidity al reproducir la música. Supongo que habrán bastantes circulando por Internet. El que uso actualmente, después de haber probado varios, y me ha dado mejores resultados, es [éste](#), enlazado al final de la [página oficial de la librería SDL_mixer](#).

No obstante, también puede obtenerse rápida y cómodamente usando Wget:

```
$ wget http://www.libsdl.org/projects/SDL_mixer/timidity/timidity.tar.gz
```

Una vez tengamos el fichero de sonidos a buen recaudo en nuestro ordenador, lo copiamos a `/usr/local/lib`, y lo descomprimos e instalamos:

```
$ su
Password:
# cd /usr/local/lib
# mv /home/usuario/timidity.tar.gz .
# tar -zxvf timidity.tar.gz
```

Esto crea un directorio llamado `timidity` dentro de `/usr/local/lib`, con todo el contenido necesario para funcionar. Sí, incluido un archivo `timidity.cfg` hecho a medida —en versiones viejas de este tutorial, usando otros ficheros de sonidos, había que escribirlo desde cero o buscarse la vida—.

Por último, es posible que necesitéis una copia —o un enlace simbólico— de este archivo en el directorio `/usr/local/share/timidity`. En mi caso lo solucioné de esta manera:

```
# cd /usr/local/share
# ln -s /usr/local/lib/timidity timidity
```

Y ahora sí, hemos terminado... Reiniciad el ordenador para asegurarnos de que todo funciona correctamente, y a disfrutar.

6. Créditos

La primera versión de este manual fue escrita por **Eye del Cul** en julio de 2006, coincidiendo con la publicación de la versión 2.1.1 de Zdoom. Desde entonces se han hecho los cambios imprescindibles para mantenerlo más o menos actualizado. Los últimos, en enero de 2014 —¡ocho años ya!—.

Podéis distribuir y modificar este documento a vuestro antojo, colgarlo en cualquier página web o publicarlo por partes... Eso sí, sería un detalle por vuestra parte que, de hacerlo, mencionarais la fuente ([Arcades 3D](#)) y a los autores originales; aunque bueno, siendo realistas, nadie os va a obligar. Todo queda en vuestra buena voluntad.

Versión en PDF creada con L^AT_EX.

¡Visita *Arcades 3D*!