Campana-Visual Mini-COMO

Alessandro Rubini, `rubini@linux.it`
Traducido por Iosu Santurtún, `sorgina@sidam.es` v2.1, Agosto de 1996. Fecha de la traducción 3 de Junio de 1998

Este documento muestra el uso de `termcap` para configurar una campana visual en el sistema propio y describe cómo deshabilitar las campanas sonoras al gusto de cada uno.

Índice General

1 Introducción

El manejador de consola de Linux emite un sonoro pitido siempre que se produce la salida de un caracter `BEL` (código ASCII 7). Aunque ésta es una buena elección para el comportamiento por defecto, a determinados usuarios no les agrada que el computador les pite (N. del T.: a mi, por ejemplo).

Este miniCOMO está orientado a explicar cómo decirles a las aplicaciones que no produzcan el código `BEL` como salida. También se proporcionan indicaciones para enseñar al núcleo y al sistema X a evitar el pitido cuando se produce la salida de un `BEL`. La mayor parte del documento se refiere a la consola de texto, ya que la configuración de este aspecto en un servidor X es sencilla para cualquier usuario que trabaje en un entorno gráfico.

En mi opinión, la mejor solución para enfrentarse a un computador ruidoso es arreglar el *hardware*, así que mi computador ni siquiera lleva altavoz.

2 *Altavozectomía*

La *altavozectomía* es con mucho, la solución más brillante para la campana sonora. Como su nombre indica, consiste en acabar con los pitidos extirpando el altavoz. Esta operación es directa y ni siquiera se necesita anestesia, pero también hay lugar para el refinamiento si se desea.

Normalmente los PCs están equipados con un estúpido interruptor para cambiar el reloj de la placa a una velocidad inferior. Este interruptor es innecesario cuando se trabaja en un entorno multitarea, ya que no es necesario enlentecer el computador para ejecutar juegos basados en bucles de *software*.

Desafortunadamente no se puede usar el interruptor para aumentar la velocidad del procesador, pero podemos convertirlo en un interruptor que habilite/deshabilite el altavoz. A veces el altavoz es útil incluso aunque se desee una máquina machaca-numeros silenciosa, por ejemplo, para indicar el final de una lenta compilación.

Para modificar la funcionalidad del interruptor, únicamente sepárelo de la placa madre y conecte sus hilos en serie con el altavoz.

Los poseedores de portátiles, desafortunadamente, no tienen fácil acceso al altavoz, y tampoco tienen un interruptor libre para dedicarlo a otra tarea. La solución preferible para tales usuarios es intentar configurar su *software* para evitar el pitido.

3 Configuración del pitido en consola

A partir de Linux 1.3.43, Martin Mares añadió la opción de configurar la campana sonora en `console.c`. Se puede cambiar la duración y el tono del pitido independientemente en cada consola, escribiendo secuencias de escape en la `tty`. Se puede guardar una configuración personalizada en el propio `~/.profile/` o `~/.login/`, para tener un pitido diferente (o no tener pitido) asociado a cada consola.

Las secuencias de escape funcionan de la siguiente manera:

- `ESC-[10;xx]` selecciona la frecuencia de la campana en Hercios. El valor debe estar en el rango 21-32766, de otro modo el resultado es impredecible (por lo menos hasta la versión 2.0.x – no puedo predecir el futuro). Si falta el argumento 'xx', se aplica el valor por defecto (750 Hz), como en `ESC-[10]`.
- `ESC-[11;xx]` selecciona la duración de la campana en milisegundos. Si se especifican más de 2 segundos, se aplica el valor por defecto (125 ms). De nuevo, si falta el argumento 'xx' (`ESC-[11]`) se utilizará el valor por defecto.

Para seleccionar, por ejemplo, un tono de 50Hz y 1 segundo de duración, se puede probar

```
echo -e "\33[10;50]\33[11;1000]"
```

en `bash` (donde `e"` significa 'interpretar secuencias de escape').

Con `tcsh`

```
echo " 33[10;50] 33[11;1000]"
```

tendrá el mismo efecto.

Aunque no conozco ninguna versión del comando `setterm` que soporte dicha configuración, una versión futura podría soportar una opción de línea de comando para configurar el sonido de la campana.

Si utiliza Linux 1.3.43 o superior, debe estar satisfecho con las secuencias de escape, y evitará tener que seguir leyendo. Si utiliza algún núcleo antiguo (yo lo hago, en mi pequeño 386), o si desea una campana visual, siga leyendo.

4 Conceptos básicos sobre `termcap`

El fichero `/etc/termcap/` es un fichero de texto que lista las capacidades de la terminal. Numerosas aplicaciones utilizan la información de termcap para mover el cursor por la pantalla y realizar otras tareas relacionadas con la pantalla. `tcsh`, `bash`, `vi` y todas las aplicaciones basadas en `curses` usan la base de datos de `termcap`.

La base de datos representa varios tipos de terminal. La variable de entorno `TERM` selecciona el comportamiento correcto en tiempo de ejecución, nombrando una entrada de `termcap` para que sea usada por las aplicaciones.

Dentro de la base de datos, cada capacidad está aparece como un código de dos letras y una representación de la secuencia de escape real utilizada para conseguir dicho efecto. El caracter separador entre diferentes capacidades son los dos puntos (":"). Como ejemplo, la camapana sonora, con código "`bl`" ("bell", campana), se representa normalmente por el string "`bl=^G`", que indica a las aplicaciones que usen el caracter `control-G`, el `BEL` de ASCII.

Además de la capacidad `bl`, hay una capacidad `vb`, que representa la campana visual ("visual bell"). `vb` normalmente no aparece en la entrada de consola en el `/etc/termcap/` de Linux.

Las aplicaciones y librerías más modernas utilizan la base de datos `terminfo` en lugar de `termcap`. Esta base de datos utiliza un fichero por cada tipo de terminal y se encuentra en `/usr/lib/terminfo`; para evitar directorios enormes, la descripción de cada tipo de terminal se guarda en un directorio nombrado con su primera letra; la entrada `linux` es, por tanto `/usr/lib/terminfo/l/linux`. Para construir una entrada `terminfo` habrá que 'compilar' la descripción de `termcap`. Véase el programa `tic` y su página del manual.

5 Definir una campana visual

Se puede añadir la entrada para la capacidad `vb` en el fichero `termcap` propio. Dennis Henriksen, `duke@diku.dk` sugirió insertar la siguiente línea en la entrada 'linux' del fichero `termcap` (en las distribuciones antiguas esta entrada se conoce como 'console'):

```
:vb=\E7\E[?5h\E[?5l\E[?5h\E[?5l\E[?5h\E[?5l\E[?5h\E[?5l\E8:\
```

La última barra se utiliza para escapar el caracter de nueva línea en la base de datos. El código de Dennis hace lo siguiente (según sus propias palabras):

- Salvar la posición del cursor (precaución de seguridad).
- Cambiar el color de fondo varias veces entre normal y video inverso.
- Restaurar la posición del cursor.

6 Deshabilitar la campana sonora en la consola de texto

Si quiere forzar una campana visual en su consola, puede reemplazar la entrada "`bl`" en el `termcap` con la misma cadena literal sugerida para "`vb`" antes. Este enfoque puede evitarle el tener que adecuar cada aplicación. Yo me decanto por esta opción en todas las máquinas en las que puedo usar Linux y no puedo quitar el altavoz.

7 Avisar a las aplicaciones

Esta es una lista incompleta de aplicaciones a las que se puede obligar a usar la entrada `vb` para el tipo de terminal actual (ya sea usando la información de `termcap` o la de `terminfo`):

- el servidor X: utilize el comando '`xset b`' para seleccionar el comportamiento de la campana. Este comando tiene 3 argumentos numéricos: volumen, tono y duración. '`xset -b`' deshabilita la campana. La configuración del servidor X afecta a todas las aplicaciones que se ejecuten en la pantalla.
- `xterm`: `xterm` puede convertir cada pitido en una señal visible o audible. Si utiliza la campana sonora, se aplicarán los valores de `xset`. Por defecto la campana en `xterm` es sonora, pero puede usar la opción de línea de comando '`-vb`' y el recurso `xterm*VisualBell: true` para convertirla en un destello visible. Se puede cambiar la señalización visible/audible en tiempo de ejecución utilizando el menú invocado al pulsar `control+botón izq. del ratón`. Si suele utilizar X seguramente no necesite la información siguiente.
- `tcsh` (6.04 y superior): '`set visible bell`'. La instrucción puede aparecer en el `.cshrc` o se puede introducir interactivamente. Para restaurar la campana sonora vale con '`unset visiblebell`'. Para deshailitar cualquier notificación introduzca '`set nobeep`'.
- `bash` (cualquier bash, por lo que yo se): ponga '`set bell-style visible`' en su `~/.bashrc/`. Los valores posibles para `bell-style` son '`none`' (niguno) o '`audible`' (sonoro).
- `bash` (con readline, así como otras aplicaciones basadas en `readline`): ponga '`set prefer-visible-bell`' en `~/.inputrc/`.
- `nvi` y `elvis`: ponga '`set flash`' en `~/.exrc/` o introduzca '`:set flash`' interactivamente (ojo con los dos puntos). Para deshabilitar la campana use '`noflash`' en lugar de '`flash`'.
- `emacs`: ponga '`(setq visible-bell t)`' en su `~/.emacs/`. Se deshabilita con '`(setq visible-bell nil)`'.
- `less`: use '`-q`' en la línea de comando para utilizar la campana visual, use '`-Q`' para deshabilitar cualquier acción. Las opciones por defecto se pueden añadir en su variable de entorno `LESS`.
- `screen`: utilice la combinación CtrlA-CtrlG. Funciona en todas las pantalla virtuales. Vea la página del manual en la sección `customization` para los valores por defecto.

8 El lado oscuro del problema

La mala noticia es que no todas las aplicaciones se basan en `termcap` o en `terminfo`. La mayoría de los programas pequeños tienen la secuencia `"\a"` (alarma) en su código fuente C. Estos caracteres se convierten en el `ASCII BEL` en el binario ejecutable. Las aplicaciones reales no suelen caer en este fallo, pero cuidado con los novatos en C que le dan sus propios programas; los estudiantes de informática son los peores, garantizado.

La única manera de hacer callar el altavoz con estas aplicaciones es la altavozectomía, o usar las secuencias de escape de Martin Mares.

9 Anexo: El INSFLUG

El *INSFLUG* forma parte del grupo internacional *Linux Documentation Project*, encargándose de las traducciones al castellano de los Howtos (Comos), así como la producción de documentos originales en aquellos casos en los que no existe análogo en inglés.

En el **INSFLUG** se orienta preferentemente a la traducción de documentos breves, como los *COMOs* y *PUFs* (**P**reguntas de **U**so **F**recuente, las *FAQs*. `:)`), etc.

Diríjase a la sede del INSFLUG para más información al respecto.

En la sede del INSFLUG encontrará siempre las **últimas** versiones de las traducciones: `www.insflug.org`. Asegúrese de comprobar cuál es la última versión disponible en el Insflug antes de bajar un documento de un servidor réplica.

Se proporciona también una lista de los servidores réplica (*mirror*) del Insflug más cercanos a Vd., e información relativa a otros recursos en castellano.

Francisco José Montilla, `pacopepe@insflug.org`.