

Indo além do beep nos seus Shell Scripts

Som na Concha!

No desktop há muitos recursos para chamar a atenção do operador, desde informações visuais na tela até a execução de sons elaborados pelo sistema multimídia, mas nos servidores nem sempre podemos contar com esses luxos. Um sinal sonoro pode ser um recurso muito útil em seus scripts, seja para indicar a necessidade de uma interação com o usuário, apontar que uma tarefa foi completada ou registrar um erro.

POR AUGUSTO CAMPOS

A maneira mais simples de gerar som em um script shell é ativando o “bipe” do console – por exemplo, através do comando `echo -e \\a`. Mas às vezes ele é um sinal insuficiente: tem pouca variação e o controle de tom e duração é complexo, além de frequentemente ser completamente desativado ou filtrado por emuladores de terminal modernos. O outro extremo da balança são os players de formatos elaborados de música – não é difícil tocar um arquivo *OGG Vorbis* ou *WAV* a partir de um script, mas há muitas razões para não fazê-lo, a começar pela dependência da placa de som.

O ponto de equilíbrio, como de hábito, pode estar no meio do caminho: com a instalação de alguns utilitários, torna-se fácil controlar o alto-falante interno, presente em praticamente todos os PCs, para produzir melodias simples – e aí é só se acostumar a trocar as fitas de backup quando tocar uma Fuga de Bach e a revisar a ocupação de espaço em disco no servidor de email sempre que ouvir a melodia de Contatos Imediatos de 30. Grau.

O utilitário clássico para gerar melodias no Linux é o *beep* [1], capaz de controlar tom e duração e de gerar seqüências de notas. A sintaxe básica do beep define os tons através da frequência

(em hertz). Para gerar uma seqüência de 8 beeps curtos e agudos que eu consigo ouvir até com a porta fechada, uso o seguinte comando:

```
beep -f 2000 -l 50 -r 8
```

Onde *f* é a frequência, *l* é a duração da nota e *r* é o número de repetições.

Compilar o beep é bastante simples, mas é bom saber que a versão mais recente é de março de 2002. Você pode encontrar problemas se tentar executá-lo como usuário não-root em terminais que não sejam o console (por exemplo, em um *xterm* ou uma sessão *ssh*) -- devido à maneira como o Linux gerencia as permissões de acesso aos dispositivos. O autor do beep considera seguro usar o utilitário em modo *suid* para evitar esse problema, mas só você pode decidir.

Entretanto, devido ao dinamismo do código aberto, já há uma alternativa: o *PCSpk*, construído a partir do ponto em que o desenvolvimento do beep estacionou. A compilação também é bastante simples (e há instruções!), e o fato de o sistema rodar como um *daemon* (chamado *pcspkd*) controlado por um utilitário (o *pcspk*) pode facilitar sua vida, até mesmo

porque o daemon não precisa ser executado por um usuário privilegiado – basta rodá-lo a partir do console. O utilitário, naturalmente, pode ser executado por qualquer usuário em qualquer terminal.

A sintaxe avançada do *PCSpk* (suportada também pelo beep, caso você prefira) permite definir as notas da forma como qualquer violonista de fim de semana está acostumado: *A* é Lá, *B* é Si e assim por diante. Meus conhecimentos musicais param no básico, mas em alguns minutos de experimentação consegui chegar a um comando que toca um trequinho da melodia de Contatos Imediatos:

```
echo "140 D54 E54 C44 C44 G41" | pcspk -n
```

A seqüência é simples: o primeiro número indica o tempo e as demais cadeias de três caracteres indicam sempre uma nota, a oitava e a duração. Músicos amadores encontram maiores explicações no arquivo *README* e alguns exemplos (inclusive uma Fuga de Bach) em arquivos anexos ao pacote. Música, Maestro! ■

INFORMAÇÕES

[1] beep: www.johnath.com/beep

[2] PCSpk: przemoc.kgb.pl