The Linux Sound Playing HOWTO

Yoo C. Chung, wacko@power1.snu.ac.kr

v1.1, 18 May 1996

This document lists applications for Linux that play various sound formats.

Contents

1 Introduction

This is the Sound Playing HOWTO. It lists the many sound formats and the applications that can be used to play them. It also lists some hacks and advice on using these applications. There are also some other interesting applications related to sound not directly related to playback. However, this document does *not* describe how one can setup a Linux system for sound support. Refer to the Linux Sound HOWTO by Jeff Tranter for instructions on setting up a Linux system for sound support and the supported sound hardware.

This deals with normal user sound applications. That is, it is only concerned about what the average user needs to know on the application side of sound, not exotic stuff like speech synthesis, or hardware stuff which is dealt in the Sound HOWTO.

1.1 Copyright of this document

1.2 Copyright of the listed applications

If there is no mention of any copyright, then the application is under the GNU General Public License.

1.3 Where to get this document

You can find the most recent version of this document on the World Wide Web at `http://plaza.snu.ac.kr/ wacko/howto/Sound-Playing-HOWTO.html`. This document can also be obtained from the *Linux Documentation Project* `<http://sunsite.unc.edu/mdw/linux.html>`, albeit it might possibly be out of date by a few weeks.

A Korean version of this document (rather outdated) is available at `http://plaza.snu.ac.kr/ wacko/howto/Sound-Playing-HOWTO.ks`.

1.4 Feedback

I am not omniscient, and I don't use all the applications in here (a few I can't even try), so there are bound to be mistakes. Also, programs usually continuously evolve, so documentation tends to get out of date. Therefore, if you find anything wrong, please send me any corrections. Suggestions or additions to this document are welcome, too.

1.5 Acknowledgments

All the authors of the applications in this HOWTO. Also, Hannu Savolainen for the great sound driver and Linus Torvalds for the great underlying OS.

Raymond Nijssen (`raymond@es.ele.tue.nl`) and Jeroen Rutten (`jeroen@es.ele.tue.nl`) for information and help on Layer 3 Shareware Encoder/Decoder.

Antonio Perez (`aperez@arrakis.es`) for information on MikMod.

2 Playing Various Sound Formats

There are many kinds of sound formats (wav, MIDI, MPEG etc.). Below, we list the various formats and the applications that can be used to play them.

2.1 MIDI

MIDI stands for Musical Instrument Device Interface. MIDI files usually have the extension `.mid`. They contain sequencing information, that is, information on when to play what instrument in what way, etc. Depending on your hardware (and maybe the software you use to play them), the sound might be awesome, or it might be downright crappy.

2.1.1 The adagio package

This package includes `mp` (a command-line MIDI file player) and `xmp` (an XView based MIDI file player). You will need the SlingShot extensions to use `xmp`. It also contains other programs for playing Adagio scores (I don't know much about this).

If you have a GUS, `mp` can also play MOD files (see section 2.2 for more information on modules).

One little annoying bug (as of version 0.5) is that the sound breaks at the end. Namely, instead of ending the sound the way the MIDI file specifies, it ends by playing the note right before the last one in a long interval. It hasn't stopped me from using `mp`, but it might prevent someone from using it for 'real' work. It also starts up relatively slowly.

The package does not mention any copyright (at least none that I can find), so I assume it can be freely redistributed and modified. Don't hold me to my word, though.

It is a port of the CMU MIDI Toolkit to Linux (well, there was enough added to make this questionable) by Greg Lee (`lee@uhunix.uhcc.hawaii.edu`).

It can be found at `ftp://tsx-11.mit.edu/pub/linux/packages/adagio05.tgz`. The binaries included here are in a.out format (linked with ancient libraries), and the `xmp` binary segfaults in a X11R6 environment (XFree86 3.1.1, libc 4.7.2). The `mp` binary works fine.

You will need a bit of hackery to compile it. Actually, it's not much of a hackery. All you have to do is to include the `-lfl` switch at the end of `SHROBJ` and `XMPOBJ` in the Makefile. This is to link in the `flex` library, which is not linked in by default. Then follow the installation instructions. And don't forget to have XView and the SlingShot extensions installed if you want to compile `xmp`.

2.1.2 timidity

Some people recommend this *experimental* program because of good sound quality (which is very true, it's much better than `mp` on a Sound Blaster 16, though probably won't be much different on a GUS). However, it suffers from high CPU loads. It plays MIDI by first converting MIDI to WAV and then plays the WAV (you can also convert a MIDI file to a WAV file without playing if you want). This is the reason for its CPU intensive nature.

It also has an optional ncurses, SLang, or Motif interface. As far as I know, a statically linked `timidity` binay with the Motif interface is not available.

You need Gravis Ultrasound patch files to use this. A small collection of GUS patch files can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/timidity-lib-0.1.tar.gz`. You'll want to find much more if you want to properly use `timidity`. Look into the FAQ included with `timidity` for more information. `timidity` itself can be found at `ftp://sunsite.unc.edu//pub/Linux/apps/sound/timidity-0.2h.tgz`.

It is written by Tuukka Toivonen (`titoivon@snakemail.hut.fi`).

2.1.3 playmidi

This is a MIDI player that plays to FM, GUS, and external MIDI. It is supposed to have a faster startup time compared to other MIDI players. It is also able to play Creative Music Files (CMF), Microsoft RIFF (RMI) files, and large MIDI archives from games such as Ultima 7.

It has an X interface and also a SVGA interface. It also has an option for real time playback with tracking all the notes on each channel and the current playback clock (included automatically with `xplaymidi` and `splaymidi`). You should do something like

```
$ splaymidi foo.mid; stty sane
```

if you are going to use the SVGA interface, since it doesn't reset the terminal tty mode properly. The SVGA interface may be removed in the near future.

It is written by Nathan Laredo (`laredo@gnu.ai.mit.edu` or `laredo@ix.netcom.com`).

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/playmidi-2.3.tgz`.

2.2 Modules

Modules (in computer music) are digital music files, made up of a set of samples and sequencing information, telling the player when to play which sample (instrument) on which track at what pitch, optionally performing an effect like vibrato, for example.

An advantage it has over MIDI is that it can include almost any kind of sound (including human voices). Another is that it sounds just about the same on any platform, because the samples are in the module. A disadvantage it has is that it has a much larger file size compared to MIDI. Another one is that it has no real standard format (the only 'real' one is the ProTracker, which many modules aren't quite compatible with). It originated on the Amiga.

They usually have the extension `.mod`. There are many other extensions depending on what format they are in.

2.2.1 tracker

This very portable program (it has been ported to many platforms) plays Soundtracker and Protracker music modules. It uses 16 bit stereo output, and I consider the quality to be very good. Use the `-mono` option if you need a simple way to reduce CPU load.

This is a giftware program (quoting the author). It is by Marc Espie (`Marc.Espie@ens.fr`).

A version of this with the Makefile already tweaked for Linux is at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/tracker-4.3-linux.tar.gz`.

2.2.2 s3mod

This plays 4/6/8 track MOD modules and Scream Tracker 3 modules. It uses 8 bit mono output with a sampling rate of 22000 Hz by default. You can use the option `-s` to enable stereo, `-b` to enable 16 bit output, and `-f` to set the sampling frequency. However, the sound output is worse than tracker (some noise), so I recommend using `tracker` instead of `s3mod` for playing ordinary MOD files (unless you have an underpowered machine). It has a much smaller CPU load compared to tracker.

It is copyrighted by Daniel Marks and David Jeske (`jeske@uiuc.edu`), but you can do anything you want with it (except that you can't claim you wrote it).

It can be found at `ftp://sunsite.unc.edu/pub/linux/apps/sound/players/s3mod-v1.09.tar.gz`.

2.2.3 gmod

This is a music module player for the Gravis Ultrasound card. 4/6/8 channel MOD, 669, MultiTracker, UltraTracker, and S3M are the supported formats.

It requires version 2.90-2 or later sound driver. And a GUS, of course. You may have to modify the kernel a little to make volume control work as you like.

This has an X interface. You need libfwf (which can be obtained from `ftp://ftp.let.rug.nl/pub/FWF` in source form, or from `ftp://sunsite.unc.edu/pub/Linux/X11/devel` for a shared ELF library) in addition to a reasonable X installation.

This can be freely distributed. It was originally written by Hannu Savolainen, and now maintained by Andrew J. Robinson (`robinson@cnj.digex.net`).

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/gmod+x-2.2.3.tgz`.

2.2.4 mod

This *beta* program plays MODs (15/31-instrument, up to 32 voices), MTMs, ULTs and S3Ms on the Gravis Ultrasound card. It has an ncurses interface. It can also use packed modules if you have `gzip`, `lharc`, `unzip`, and `unarj` installed. It cannot play Powerpacked modules or modules packed with some Amiga composers ("PACK" signature).

This requires at least version 3.0 of the sound driver. It *won't* work with the 2.90-2 version of the sound driver included in the 1.2.x kernels.

The text interface requires ncurses. There is also an X interface included, which uses Tcl/Tk.

It is written by Mikael Nordqvist (`mech@df.lth.se` or `d91mn@efd.lth.se`).

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/mod-v0.81.tgz`.

2.2.5 pmod

`pmod` plays XM, S3M, MTM, MOD, and 669 modules on the Gravis Ultrasound card. It uses its own beta sound driver that comes with it. It claims that you can play modules and play the Linux version of Doom at the same time if you have a GUS MAX.

It has a text interface that uses ncurses.

The author is Jaroslav Kysela (`perex@pf.jcu.cz`). `pmod` also has it's own mailing list (`pmod@pf.jcu.cz`), which I think is the preferred place to ask questions about `pmod`.

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/pgmod-1.25e.tgz`.

2.2.6 MikMod

This portable module player plays XM, ULT, STM, S3M, MTM, MOD and UNI formats. (The UNI format is an internal format used by MikMod.) It uses 16 bit stereo for the sound output. Use the `-m` option (for mono output) if you need to lower the CPU load.

It was written by Jean-Paul Mikkers (`mikmak@via.nl`). This is shareware that has to be registered if you want to use it commercially. You also need permission to redistribute it commercially (non-commercial redistribution is does not need such permission).

It can be found somewhere below `/pub/demos/` at `ftp.cdrom.com` or somewhere below `/pub/msdos/programming/` at `teeri.oulu.fi`. Or you could get it from the *author's homepage* `<http://www.stack.urc.tue.nl/ mikmak/welcome.htm>`.

2.2.7 nspmod

This is an *alpha* module player which can play MTM, S3M, and MOD modules. It is intended to be a module player for soundcards without a DSP (not to be confused with what Creative Labs calls a DSP). It has a CPU load somewhat similar compared to `tracker`.

It has a feature which lets modules loop if they want to. The number of loops can be limited by the `-l` option. It uses only 8 bit sound output (as of version 0.1).

This was written by Toru Egashira (`toru@jms.jeton.or.jp`).

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/nspmod-0.1.tar.gz`.

2.2.8 yampmod

This *alpha* program was designed to play 4-channel modules using the minimum of CPU resources. It was *not* designed to produce high quality sound. So the only sound output it produces is 22 kHz mono output. Also, the output isn't as clean as it should be, reflectng its alpha status.

It was written by David Groves (`djg@djghome.demon.co.uk`).

It is available at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/yampmod-0.1.tar.gz`.

2.3 MPEG audio streams

MPEG is a standard specifying the coding of video and the associated audio for digital storage. MPEG is usually associated with video, but the audio part of the standard can be used separately. The audio part of the MPEG standard defines three layers, layer I, II, and III. Players that can decode higher layers can also decode lower layers (e.g. layer III players can play layer II files). Layer I MPEG audio files usually have the extension `.mpg` (so if there is a file with this extension that can't be played by a MPEG video player, it's probably an audio stream), layer II usually have `.mp2`, and layer III usually have `.mp3`. The files are usually rather large compared to other formats, but the quality is very good. A two megabyte layer II MPEG audio file will probably take up 25 megabytes for a raw PCM sample file with the same quality.

2.3.1 maplay

This MPEG audio stream player only has support for layer I and layer II streams. It lacks support for layer III as of version 1.2. It supports 16 bit sound cards on Linux (though some 16 bit sound cards are rumored not to work with `maplay`).

It is *very* CPU intensive, taking up to about 55% CPU time on a 60MHz Pentium. The output is intolerable on a 66MHz 486 because the CPU just can't catch up with the sound. If this happens to you, try playing only one side of the audio stream (with the `-l` or `-r` option), instead of the default stereo.

A slight change in one of the files may be necessary in order to compile it. Namely, you may need to add the following line to the beginning of the file `configuration.sh`.

```
#! /bin/sh
```

The author is Tobias Bading (`bading@cs.tu-berlin.de`).

`maplay` can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/maplay1_2.tar.gz`.

2.3.2 Layer 3 Shareware Encoder/Decoder

This is actually a converter that converts MPEG Layer 3 audio streams to WAV, AIFF, SND, AIFC, or just raw PCM sample files. The Linux version does not directly output the sound to the soundcard. One has to first convert it to some other format.

However, when you try to play a converted file using `sox`, you'll probably just get noise because the word order in the PCM samples is not right (at least on Intel platforms). You need to give `sox` the option `-x` to solve this problem. But there are some players that don't have to be told that the word order is wrong, so you might not have to worry about this.

If you have a really fast computer (probably at least a 90Mhz Pentium), then you can try to play it directly without having to first convert the MPEG audio file to another format like in the following example (this example assumes that you're using `sox`).

```
$ l3dec foo.mp3 -sto | play -t raw -x -u -w -c 2 -r 44100 -
```

Of course, the number after `-r` should be the sample rate of the audio stream, and the number after `-c` depends on whether it is mono or stereo (or even quad). If this looks too complicated, you can use something like a shell script or an alias.

This is shareware copyrighted by Fraunhofer-IIS. A demo version is available at `ftp://ftp.fhg.de/pub/layer3/l3v200.linux.tar.gz`. The demo version only converts layer III audio streams.

2.4 WAV

Quote from the `sox` man page:

> These appear to be very similar to IFF files, but not the same. They are the native sound file format of Windows 3.1. Obviously, Windows 3.1 is of such incredible importance to the computer industry that it just had to have its own sound file format.

These usually have the extension `.wav`.

Also see section 2.5.1 for another WAV player besides the ones listed here.

2.4.1 wavplay

This *unfinished* program has a command line interface and an X Window interface for playing and recording in WAV format.

Copyright is not explicitly mentioned. It is by Andre Fuechsel (`af1@irz.inf.tu-dresden.de`).

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/wavplay021.tar.z`. A patch for systems with sound drivers higher than 2.0 (which means almost all systems these days) is available at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/wavplay021_patch.tar.gz`.

2.4.2 XWave

This program is based on `wavplay`. It too has an X Window interface for playing and recording in WAV format. However, it also has some editing capabilities and can add effects.

This is still in a rather buggy stage, so the author would appreciate any bug reports, bugfixes, or new code.

This is registered freeware, in other words you can freely redistribute it as long as you register it by sending the author a color picture postcard. But it's OK to use it unregistered, as long as you don't get caught???

The author is Will Fish (`fishwj@ee.port.ac.uk`).

XWave can be found at `ftp://ftp.ee.port.ac.uk/pub/linux/sound/xwave0.01-1.tar.gz`.

2.5 Other stuff

2.5.1 sox

This program is actually a converter, that is, it converts one sound format to another. When invoked as `play`, however, it plays the sound (the `play` application in the Sound HOWTO probably refers to this). It supports raw (no header) binary and textual data, IRCAM Sound Files, Sound Blaster `.voc`, SPARC `.au` (w/header), Mac HCOM, PC/DOS `.sou`, Sndtool, and Sounder, NeXT `.snd`, Windows 3.1 RIFF/WAV, Turtle Beach `.smp`, CD-R, and Apple/SGI AIFF and 8SVX formats (a lot of these formats I have no experience with, I just took it from the man page).

Since somewhere in the 1.3.6x kernels, you might have to make a small change in one file to make it play the sound directly. Namely, you may have to change line 179 in sbdsp.c from

```
if (abuf_size < 4096 || abuf_size > 65536) {
```

to

```
if (abuf_size < 1 || abuf_size > 65536) {
```

But then again, you might not have to do this. But doing this won't break anything.

It is written and copyrighted by many people, and can be used for any purpose.

It can be found at `ftp://sunsite.unc.edu/pub/Linux/sound/convert/Lsox-linux.tar.gz`.

2.5.2 cat

One might think what `cat`, the sometimes overused concatenating utility, has to do with playing sounds. I'll show a use of it through an example.

```
$ cat sample.voc > /dev/dsp
$ cat sample.wav > /dev/dsp
$ cat sample.au > /dev/audio
```

Doing a `cat` of an `.au` file to `/dev/audio` will usually work, and if you're lucky enough that the file has the correct byte order (for your platform) etc., a `cat` of a sound file that uses PCM samples (like `.wav` or `.voc`) to `/dev/dsp` might even sound right.

This isn't a totally useless use of `cat`. It might be useful, for example, if you have a sound file that none of your programs recognize, and you know that it uses PCM samples, then you might be able to get a very approximate idea on how it sounds like this way (if you're lucky).

3 Other useful sound utilities

This section has nothing to do with the actual playing of sound files. Rather, it is a collection of some sound utilities that one might find useful.

3.1 volume

This is a simple command line interface for controlling the volume (what else could it be?). It also has a separate program with a Tcl/Tk interface included in the package for controlling the volume and playing `.au` sound files.

It is very useful for people who don't have hardware volume control, don't want to pump in extra electricity for the external amplifiers (which might degrade the sound output, unless you have a good one), don't have much need for *really* loud sounds, and want a convenient command line interface (I'm a command line geek) for controlling the volume.

This is Freeware and it is written by Sam Lantinga (`slouken@cs.ucdavis.edu`).

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/volume-1.1.tgz`.

3.2 setmixer

This is a command line mixer control program.

The author is Michal Jaegermann (`michal@ellpspace.math.ualberta.ca`).

3.3 Tickle Music

This *beta* Tcl/Tk program is a music file browser that allows you to play various sound formats as long as an appropriate program to play it is on your system. By default `gmod` is used for playing MOD files and `mp` for playing MIDI files (you can change the source to use another program).

It is written and copyrighted by Shannon Hendrix (`shendrix@pcs.cnu.edu` or `shendrix@escape.widomaker.com`).

It can be found at `ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/tmusic-1.0.tar.gz`.

4 References

1. The documentation included with the applications in this document.
2. The Linux Sound HOWTO. It is available from the *Linux Documentation Project* <http://sunsite.unc.edu/mdw/linux.html>.
3. *The Linux MIDI and Sound Pages* <http://www.digiserve.com/ar/linux-snd/>
4. `alt.binaries.sounds.mods` FAQ
5. *MPEG Audio Layer 3 FAQ* <http://www.iis.fhg.de/departs/amm/layer3/sw/index.html>