The Linux MP3 HOWTO

By Philip Kerr, phil@websentric.com v1.10, September 1999

This document describes the hardware, software, and procedures needed to encode and play *MP3* sound files under Linux.

Contents

1 Introduction.

This document describes the hardware, software and procedures needed to encode and play MP3 sound files under Linux.

2 Copyright of this document.

3 Where to get this document.

The most recent official version of this document can be obtained from the Linux Documentation Project `<http://metalab.unc.edu/LDP/>`.

4 Acknowledgments.

In writing this HOWTO I have had to draw heavily on the *Sound-HOWTO* By Jeff Tranter, and the *Sound-Playing-HOWTO* By Yoo C. Chung.

Thanks also to the other HOWTO authors whose works I have referenced:

Linux System Administrators Guide By Lars Wirzenius.

Linux Network Administrators Guide By Olaf Kirch.

Multi Disk System Tuning HOWTO By Stein Gjoen.

5 Disclaimer.

Use the information in this document at your own risk. I disavow any potential liability for the contents of this document. Use of the concepts, examples, and/or other content of this document is entirely at your own risk.

All copyrights are owned by their owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark.

Naming of particular products or brands should not be seen as endorsements. You are strongly recommended to take a backup of your system before major installation and backups at regular intervals.

6 Hardware Requirements & Performance Issues.

Digital Audio processing is a resource intensive task that relies heavily on the processing and I/O capabilities of a system. I would strongly recommend a Pentium class machine as a minimum.

If you are going to be encoding from an analogue audio source, a PCI soundcard will give the best results. The I/O performance difference between an ISA and PCI based card is significant, over 132 Mbits/sec for PCI (quote taken from the *PCI-HOWTO*). Naturally, the better the quality of the soundcard in terms of its signal-to-noise ratio, the better the encoded MP3. I've been using the Soundblaster PCI128 and just switched over to a Soundblaster Live Value; both cards give good audio performance, but the Live has significantly better S/N ratios, good enough for semi-pro audio work. Remember the old data processing maxim:- garbage in - garbage out!

When recording analogue audio to a hard disk, more commonly referred to as direct to disk or d2d recording, the performance of the disk, and its interface is critical. If you are using an IDE based based system, mode 4 or UDMA is preferable as they have a semi-decent transfer rate.

The ideal solution would be to use a SCSI based system as the drives and interface have far better throughput capabilities, a sustained 5mbits/sec for SCSI 1 through to 80mbits/sec for ultra/wide SCSI. IDE can peak at anything from 8.3 MB/s to 33 MB/s for Ultra-ATA but these speeds are peak, average transfer rates will be slower. If you can find, or afford, an AV SCSI drive, go for it. AV drives have had the read/write head system optimised for continuous data transference; other SCSI and IDE drives normally cannot sustain continuous data transfer as the write head heats up!

Naturally a drive that has cache will give more consistent results than one that doesn't, as the cache will act as a buffer if the heads do lift or it cannot handle the throughput.

If your drive isn't up to spec, your recording will suffer from dropouts and glitches, where the drive failed to record the signal. If you are recording one-off events, such as live performances invest in a good SCSI based disk system.

Another cause of d2d dropouts is a heavily loaded system. Background tasks can cause the system to momentarily glitch. Its recommended to run as few background services as you can, especially networked based services. For more information about setting network services, and startup scripts please refer to the *SAG* and *NAG* guides.

Virtual memory paging will also cause glitches, so run with as much physical RAM as you can, I'd recommend at least 32 Mb, but you may well need more.

For those wanting to extract the most out of their system, optimising the kernel probably wouldn't do any harm either.

While the hardware specifications above will give you a decent system to encode audio data, don't discount using older, lower spec kit if that's all you have access to.

It'll be a good challenge for a sys-admin to tweak a low-spec system to give good results, and the end result will probably be a happier Linux box.

Another important issue is the audio cabling. Cheap, poor quality cables and connectors will result in poor recording quality. If your soundcard has the option to use phono, sometimes referred to as RCA, connectors, use them. Gold plated contacts will also help maintain audio quality, as will keeping audio cables away from data cables as there will be a chance of interference between them.

But don't forget, spending a fortune on the best audio cabling will be lost if the rest of the system hasn't been optimised.

For encoding mp3's from CD-ROM, the speed or type of drive will determine the time taken to read the raw information from it. A single speed drive will probably be too slow for all but the most patient.

Your CD-ROM must be connected to your soundcard, either using the internal connector, or by taking the headphone's output and patching it into the line-in on the soundcard.

For detailed instructions on setting up soundcards, now would be an excellent time to read the *Sound-HOWTO*.

7 Software Requirements.

Converting audio to MP3's is normally a 2 stage process, first the audio is recorded into a WAV format, then the WAV is then converted into an MP3. Depending on what format you wish to encode audio from, CD or direct audio, will determine what software tools you need to produce the WAV file.

If you are wanting to encode from audio input, you will need a program that will record from your soundcard's input and save the results in a WAV format. Below are some useful utilities.

Wavrec

Wavrec is distributed as part of wavplay, which can be downloaded from:-

`<ftp://sunsite.unc.edu/pub/Linux/apps/sound/players/>`

To convert CD audio data to WAV format, sometimes known as CD ripping:

CDDA2WAV

`<http://metalab.unc.edu/pub/Linux/apps/sound/cdrom/>`

Cdparanoia

`<http://www.xiph.org/>`

To convert the WAV file, to MP3 format:

Blade's MP3 Encoder

`<http://bladeenc.cjb.net>`

To play the MP3 you will need a player:

Xmms (Formerly known as X11Amp)

`<http://www.xmms.org>`

Xaudio

`<http://www.xaudio.com>`

You may also need a mixer program; Xmixer works well and is included with most distributions.

8 Setting up your system.

This section will describe the basics of setting up your Linux system to record audio from either an analogue or CD-ROM source.

I'm basing this section around my Intel based Linux system which is running Redhat 5.2, but should be reasonably distribution neutral. I'll be working on the Sparc platform version shortly. (if you have any success in using this HOWTO on other hardware, please get in touch).

Naturally a reasonable prerequisite is a working soundcard. At this point in the HOWTO, I invite you to read the excellent *Linux Sound HOWTO*, by Jeff Tranter. After which a good read of the *Linux Sound Playing HOWTO*, by Yoo C. Chung. Both of the above mentioned HOWTO's cover the details of getting a sound system working under Linux far better than I could.

8.1 Setting up for Analogue Audio Capture.

Firstly, set up your audio. There are a multitude of ways to route audio before it gets to your Linux box, some common ones are:

Line out to Line in. Most audio devices have a Line output sockets. Line level is a standard that specifies what voltage and current the audio device will send out. If I remember correctly it is 500mA for domestic and Semi Pro devices, and 750mA for Pro audio devices. I would guess that the standard set for most soundcards will be 500mA, but some of the newer Pro audio may be to the higher standard It shouldn't make too much difference unless you are recording at very high levels.

The Line level output is normally used to connect HI-FI equipment to an amplifier, so things such as Tape Decks, Radio Tuners, CD players, DAT machines and Mini-Disc players should connect without problem. Turntables can be more of a problem, see below for more information.

You could capture audio from VCR's as well. Most VCR's will either have Line out for sound, or you can Get a Line out from a SCART socket if your VCR has one.

Amplifier Tape out to Line in, Line out to Amplifier Tape in. This configuration is essentially replacing a traditional tape recorder connected to your HI-FI amplifier with your Linux system. The Line out to Tape in allows monitoring of the recording levels.

Mike to Mike in. The voltages generated by microphones is very much smaller than those used in Line level devices. If you were to plug a Microphone into the Line in, chances are you would never record anything.

WARNING, doing the reverse, plugging a Line level device into the Microphone input, can damage your soundcard!!

Turntable to Mike in. Sadly I've no information on this at the moment. If I remember correctly the turntable should be connected to the Microphone in, it uses a similar mechanism to record sound and has similar low voltages.

Music keyboards & synths should be connected to Line in, with guitars connecting to Line in via a DI (Direct Injection, used to convert the signal to Line level) box.

Before you plug in anything into your soundcard, make sure the volume levels are turned down to minimum, or if using microphones they are either turned off or away from speakers.

8.2 Setting up for CD-ROM Audio Capture.

Setting up your Linux system to extract audio data from CD-ROM is reasonably straight forward.

If you can hear a track playing from your CD-ROM through your speakers or amplifier, connected to your soundcard, then there's a reasonable chance you should be able to record from it.

8.3 Additional Setting up.

Log in as per normal to your system, then using a mixer program set the recording levels that are loud enough to give you a decent recording level, but aren't too loud and distorting. I normally just judge this by ear, after a while you'll get to know what levels are best for your kit.

I recommend either turning off all unnecessary services or switching to the single user runlevel, especially when encoding from an audio source. This is to ensure that the bare minimum of services are running and thus minimising system glitches when recording.

I've set up a separate SCSI drive, exclusively to record the audio to, which I'll refer to as /mnt/mp3. I've done this mainly for the performance gains in using a SCSI drive. Also, recording onto a dedicated drive, where you are almost certain the head isn't going to suddenly skip to another part of the drive as you are writing audio data to it, is a good thing :)

For details on setting up a Linux system with multiple disk drives, a good read of the *Multi-Disk-HOWTO*, by Stein Gjoen may be useful.

9 Encoding from Audio.

Firstly, make sure you have enough space on your drive. At CD quality, 44.1 Khz 16 Bit stereo, 1 minute takes nearly 5 Mb.

I normally record at DAT quality, which is 48 Khz 16 Bit stereo.

Using wavrec I use the following syntax:

/usr/local/bin/wavrec -t 60 -s 48000 -S /mnt/mp3/temp.wav

The first part is an explicit path to wavrec. The '-t 60' specifies the length of time to record for, in seconds.

The third option, -s 48000 refers to the sample rate in bits/sec. (48000 is the rate for DAT, 44100 is CD)

The last option is the path to the output file.

To see the full set of options, run waverec -help, or see it's man page.

This will produce your WAV file Next you will need to encode it into MP3 format.

Use bladdenc with the following command line.

/usr/local/bin/bladeenc *[source file] [destination file]* -br 256000

The -br option sets the bit rate, in this case I've set the rate to the maximum rate of 256k bits/s. The path to bladeenc may also be different on your system to the one I've used in my example.

To see the full set of options, run bladeenc -help, actually this is an invalid option, but will display the list of options.

10 Encoding from CD-ROM.

Again, check you will have enough drive space on your system.

I've written a very simple Perl script that will rip and encode tracks from a CD.

```
#!/usr/bin/perl

if ($ARGV[0] ne "") {

$count = 1;

do {

$cdcap = system("cdparanoia", $count, "/mnt/mp3/tmp/cdda.wav");
$track = "$ARGV[1]/track".$count.".mp3";
$benc = system("bladeenc  /tmp/cdda.wav $track -br 256000");
$count++;

}
until $count > $ARGV[0];
exit;
}

else {
print "Usage cdriper [no of tracks] [destination directory]\n\n";
}
```

The main lines of interest are

$cdcap = system("cdparanoia", $count, "/mnt/mp3/tmp/cdda.wav");

This line calls the CD ripper, cdparanoia. Cdparanoia converts raw CD audio data to WAV format. The salient options are $count, which is the number of tracks to rip, and then the path for the outputted WAV file. In my example this will go to a tmp directory on my MP3 SCSI drive.

The WAV file is then converted into a MP3 file using Bladeenc.

I've written this Perl script in order to rip a CD without having to rip and encode each track, and without having to use the batch mode of Cdparanoia. This cuts down on free disk space needed as Cdparanoia's batch mode will rip the whole disk, and take up anything upto 600 Meg.

Please note: The above script is very basic and has nothing fancy, like error checking. Improve at your leisure :)

11 Playing MP3's.

So, hopefully, you should now have some MP3 files ready to play.

The two most popular MP3 players for Linux are Xaudio, and Xmms (formerly known as X11Amp).

Links to the homepages of both players can be found in section 7.

I've based this HOWTO around my use of xaudio, primarily because I can use it with or without being in X.

The program is shareware.

Xaudio is a suite of programs, of most interest to us are xaudio and mxaudio.

After installing the files onto your system, you have the choice or running either the command line version, xaudio, or the X version mxaudio.

Mxaudio has a nice interface, and is straightforward to use.

Xaudio's options can be viewed by running the program with the -help option (xaudio -h).

To play a track try xaudio track.mp3

To play a whole directory, xaudio *.mp3

Both programs have README files.

12 Feedback.

New hardware and software is being released all the time. If you are using newer versions of the hardware and / or software listed within this document, or can add to anything within this area, please send your information for inclusion to *phil@websentric.com* and I'll include it in the next release.

Happy MP3'ing!!