Linux MP3 CD Burning mini−HOWTO

Greg Wierzchowski

<greg@draxen.org>

2001−05−29

Revision History

Revision 1.11	2004−08−21	Revised by: GW
Fixed a typo.		
Revision 1.10	2004−07−14	Revised by: GW
Fixed revision numbering to be in sync with CVS.		
Revision 1.9	2004−07−14	Revised by: GW
Small updates, added new sub−section.		
Revision 1.8	2003−09−12	Revised by: GW
Cleanup, update, fixes and minor layout change.		
Revision 1.7	2002−04−26	Revised by: GW
Divided into two sections: existing material to section "audio", new section "data" on data CDs. Misc. additions.		
Revision 1.5	2001−11−19	Revised by: GW
Fixed omission in Disc−At−Once burning section.		
Revision 1.4	2001−11−17	Revised by: GW
Added Disc−At−Once burning section.		
Revision 1.3	2001−09−02	Revised by: GW
Added another example of decoding MP3 files with lame.		
Revision 1.2	2001−07−12	Revised by: GW
Minor layout changes; Added Translations subsection into Credits.		
Revision 1.1	2001−06−12	Revised by: GW
Minor cleanup; Regexp fix for MP3 to WAV name conversion example.		
Revision 1.0	2001−05−29	Revised by: GW
Initial Release.		

A complete recipe for creating audio and data CDs from MP3 files.

Table of Contents

1. Introduction

This mini−HOWTO was created because of my experience with burning music CDs and lack of some specific information about sound normalization on the Internet. I usually burn music CDs as a mix − different songs from different sources.Very often volume level between songs varies greatly. This is the first obstacle. Second, many of the files on the Internet are not CD−compatible (16 bit, stereo, 44.1 kHz) and have to be converted. There are many programs to burn music CDs from MP3 files, and many of them do the conversion transparently. But I haven't seen a single tool that also normalizes the volume, so that's why I worked out my own CD−burning recipe.

If you you just want to make a CD filled with music, and not be bothered with all of the details, I have good news for you: Kees Cook (`<kees@outflux.net>`) put together a tool based on this HOWTO, which automates all of the tasks outlined here. His program can be obtained from http://outflux.net/unix/software/mp3cd/. Thanks, Kees!

This HOWTO is just about one thing − putting MP3 music on a CD, so that you can listen to it. For in−depth information about MP3 files, please look at *The Linux MP3 HOWTO* by Phil Kerr, located at http://www.tldp.org/HOWTO/MP3−HOWTO.html. For information about CD creation in general as well as CD burners, refer to *CD−Writing−HOWTO* by Winfried TrÃ¼mper, available at http://www.tldp.org/HOWTO/CD−Writing−HOWTO.html. Also, look at the *CD Burning Resources* section for more information.

I'm assuming you wish to burn a CD with the collection of songs you obtained from different sources, all varying quality, but you want to get the best−sounding CD possible. This mini−HOWTO outlines the steps that may help you.

1.1. Copyright and License

---

2. Audio CDs

2.1. Preparing the Tracks

Note

All commands assume bash shell

2.1.1. Filename Cleanup

1. Collect all MP3 files in one directory.
2. If any filenames contain spaces, first convert them to underscores:
   ```
   for i in *.mp3; do mv "$i" `echo $i | tr ' ' '_'`; done
   ```
   This first step is important because, even if unix itself allows spaces in filenames, most programs get confused by them.
3. If your MP3 files came from DOS/Windows, they may have uppercase extensions. You can convert whole names to lowercase or just extensions. For everything lowercase do:
   ```
   for i in *.[Mm][Pp]3; do mv "$i" `echo $i | tr '[A-Z]' '[a-z]'`; done
   ```
   to convert just extensions:
   ```
   for i in *.MP3; do mv "$i" "`basename "$i" .MP3`.mp3"; done
   ```

2.1.2. Conversion

The CD Audio specification requires wave files that are 16 bit, stereo, 44.1 kHz. MP3 files often have different parameters, but we need to convert them to WAV first, anyway.

We have several choices for the conversion process.

2.1.2.1. mpg123 and mpg321

Originally, there was only **mpg123**. However, it uses a proprietary licensing, and now there's an open source replacement – **mpg321**. Both commands use the same syntax:

```
for i in *.mp3; do mpg321 -w `basename $i .mp3`.wav $i; done
```

When decoding 22khz MP3 files the output of **mpg123** may be distorted. I don't know how well **mpg321** deals with this problem. If you're converting with **mpg123**, use:

```
for i in *.mp3; do mpg123 --rate 44100 --stereo --buffer 3072 --resync -w `basename $i .mp3`.wav $i; done
```

Mpg123 can be obtained from http://www.mpg123.de/.

Mpg321 is available from http://mpg321.sourceforge.net/.

NOTE I noticed that with some MP3 files mpg123 output was distorted. At first I thought that MP3's were bad, but then I checked with another player and they sounded OK. So **mpg123** is not my converter of choice.

2.1.2.2. MAD

Another MP3 player/decoder, and the one I prefer, is **madplay**. It's available from http://www.mars.org/home/rob/proj/mpeg/. With *madplayer*, the command line is:

```
for i in *.mp3; do madplay −o `basename $i .mp3`.wav $i; done
```

Unfortunately, **madplay** also had problems with some of MP3 files I had. I don't think there's a problem with the decoder, but rather with it handling broken MP3 files.

2.1.2.3. Lame

The **lame** encoder, which has a decoding mode, seems to handle difficult cases very well (**lame** can be found at http://www.mp3dev.org/mp3/) :

```
for i in *.mp3; do lame −−decode $i `basename $i .mp3`.wav; done
```

2.1.2.4. OGG files

Well, this HOWTO is about MP3 files, but OGG standard is an alternative which many people prefer over MP3 because of the licensing. To deal wit OGG files, you need to use:

```
for i in *.ogg ; do ogg123 −d wav −f `basename $i .ogg`.wav $i; done
```

2.1.2.5. Converting other formats

Nico Genimakis sent me an email about using mplayer to covert audio in many different formats to WAV with automatic resampling to 44100Hz. Mplayer is known to be able to read almost anything, and it can convert your .ogg, .mp3, .flac, .wma etc.

usage (in this example converting WMA files):

```
for i in *.wma ; do mplayer −vo null −vc dummy −af resample=44100 −ao pcm −waveheader $i; done
```

2.1.3. Conversion Notes

NOTE: The **`basename $i .mp3`.wav** command replaces MP3 extensions with WAV. There are 101 ways to do that, here's the alternative: **`echo "$1" | sed 's/\.mp3$/.wav/'`**

After the conversion, run "**file *.wav**" and check the output for any files that differ from 16 bit, stereo 44100 Hz.

If there are files with different characteristics, convert them to the above specs. For example, to convert file track01.wav to obtain sample rate 44.1 kHz, you could use:

```
sox track01.wav −r 44100 track01−new.wav resample
```

or, if the above introduces static when converting mono files:

```
sox track01.wav −r 44100 −c 2 track01−new.wav
```

Sox is so popular, that it's probably installed by default with any Linux distribution, and can be obtained from http://www.spies.com/Sox/. However, the command−line options are somewhat cryptic for the casual user (me). Look at http://www.spies.com/Sox/sox.tips.html for some tips on usage.

2.2. Normalisation

Normalisation is a process during which all the sound files are brought to the same relative loudness level. I use a program by Chris Vaill (`<cvaill@cs.columbia.edu>`), called **normalize** − it can be obtained from http://www.cs.columbia.edu/~cvaill/normalize/

I use the following syntax (−m is for mix mode, where all files should be as loud as possible):

```
normalize −m *.wav
```

3. Burning Your CD

There are many programs to create CDs from WAV files. I use **cdrecord** for command−line burning and **XCDROAST** for gui. For **cdrecord**, you have to know what SCSI device your CD−writer is. If you're using ATAPI writer, with older kernel, use SCSI emulation (kernel module ide−scsi). As of kernel 2.6, you can use ATAPI directly, without SCSI emulation, by prepending *ATAPI:* to the device specification. Let's assume, that your ATAPI cdwriter is on the second IDE bus as a master. Thus, it will have `/dev/hdc` device file. To instruct the kernel that you want to treat it as a SCSI device, add the following line to `/etc/lilo.conf`:

```
append=" hdc=ide-scsi"
```

Also, if your kernel doesn't automatically load ide−scsi module, add **insmod ide−scsi** into your `rc.local` (or equivalent) file. Once you have our CD−writer recognized as a SCSI device, run **cdrecord −−scanbus** to find out what's the "dev" parameter to cdrecord. On my system, the output looks like the following:

```
scsibus1:
1,0,0 100) 'IOMEGA ' 'ZIP 250 ' '51.G' Removable Disk
1,1,0 101) 'HP ' 'CD-Writer+ 7100 ' '3.01' Removable CD-ROM
```

So, the **cdrecord** command line will contain **dev=1,1,0** to specify the device. Here is the complete command on my system:

```
cdrecord dev=1,1,0 -eject speed=2 -pad -audio *.wav
```

And, with kernel 2.6:

```
cdrecord dev=ATAPI:1,1,0 -eject speed=2 -pad -audio *.wav
```

NOTE

The −pad argument is neccessary, because all audio tracks on the CD must be adjusted for the proper data length, which is not always the case with mp3 files.

Another way, would be to convert WAV files with **sox** into CDR format before burning:

```
sox file.wav file.cdr
```

4. Burning a DAO CD

DAO, Disc−At−Once, is as of now the only method for burning a CD without a 2−second pause between the tracks. It's useful for burning party mixes. The program for burning CDs in DAO mode is **cdrdao**, available from SourceForge, http://sourceforge.net/projects/cdrdao/.

The **cdrdao** program uses description files called *TOC* (Table Of Contents, of course). There are two ways to create such file. First is to use a shell script, distributed with **cdrdao** source (in `contrib` directory, called **generate_toc.sh**. It takes a list of `.wav` files as an argument and produces a `cd.toc` file. Second way is to simply create such file yourself in a text editor of your choice. Here is a self−explanatory example:

```
CD_DA

TRACK AUDIO
AUDIOFILE "mix-01.wav" 0

TRACK AUDIO
AUDIOFILE "mix-02.wav" 0

TRACK AUDIO
AUDIOFILE "mix-03.wav" 0

TRACK AUDIO
AUDIOFILE "mix-04.wav" 0

TRACK AUDIO
AUDIOFILE "mix-05.wav" 0
```

The *0* (zero) after the wave filename means start from the beginning of the file. There can be a second number providing the length (time) of file to record. The **xcdroast** creates similar *TOC* files, there are also examples in `testtocs` directory of **cdrdao** source.

The **cdrdao** by default uses the device `/dev/cdrecorder`, which should be a link to the cdwriter device. Assuming your cd recorder device file is `/dev/scd0`, create the link (as root) as follows:

```
ln -s /dev/scd0 /dev/cdrecorder
```

Then, assuming that the *TOC* file is named `cd.toc` the command to burn the cd is simply:

```
cdrdao write cd.toc
```

5. Software

There are some programs available, that can automate the process of creating CDs from MP3 files. Here is arbitrarily selected list:

- *burnmp3* – Program to automate burning with DAO method. http://richardsnow.bizland.com/burnmp3/.
- *mp32dao* – a script from *cdrdao* distribution, in the *contrib* directory. http://cdrdao.sourceforge.net/.

6. Data CDs

Note

This section is a work in progress, you're looking at initial, very sparse version.

With increasing popularity of CD/MP3 players burning data CDs for listening purposes become practical. The advantage is definitely being able to squeeze ten times more music onto one CD (a very approximate figure).

First I would recommend the filename cleanup steps outlined at the beginning of this document. Conversion to WAV is, obviously, not needed here. To normalize MP3 files without losing quality by conversions, use **mp3gain**. This program is available from http://www.maazl.de/project/mp3/mp3gain.html or http://mp3gain.sourceforge.net/ (but this second site seems to contain Windows−oriented version). You'd need to issue the command:

```
mp3gain -r -c *.mp3
```

As far as burning is concerned, MP3 data CD−s are just a standard data CD's (ISO9660) with MP3 files on it. All MP3−CD players I know accept CD−s with directories in them, and I usually burn CD with Joliet extension and they work just fine. So to burn such a CD under linux, you first need to create an ISO image an then burn it on the CD as in the example below:

```
mkisofs -J -o /tmp/mymp3s.iso /home/greg/mp3s/
cdrecord dev=1,0,0 speed=16 -eject /tmp/mymp3s.iso
```

That's it!

7. CD Burning Resources

- Burning CDs on Linux: http://www−106.ibm.com/developerworks/linux/library/l−cdburn.html
- CD Burning from the command line: http://freeunix.dyndns.org:8088/site2/howto/Burn_em_Baby.shtml
- Comprehensive Burning Tutorial: http://www.yolinux.com/TUTORIALS/LinuxTutorialCDBurn.html
- CD Mastering Guide: http://debianlinux.net/~jama/howto/cd_mastering_steps.html

8. Credits

Special thanks to all the people who contribute to the Linux community and who made this HOWTO possible.

8.1. Translations

- Im Eunjea − Translated this document to Korean, URL is http://kltp.kldp.org/eunjea/mp3_burning/.
- Mendel L Chan − Translated this document to Chinese, URL is http://www.linux.org.tw/CLDP/mini/MP3−CD−Burning/.
- Chie Nakatani − Translated this document to Japanese, URL is http://www.linux.or.jp/JF/JFdocs/MP3−CD−Burning/index.html.
- Laszlo Daczi − Translated this document to Hungarian, URL is http://tldp.fsf.hu/HOWTO/MP3−CD−Burning−hu/index.html.
- Alix − Translated this document to French, available at French TLDP site http://www.traduc.org.

8.2. Other Credits

I have had a great response to the previous releases of this HOWTO. So many people contributed, that I decided to change this section slightly in order to list all of the names. So, I'd like to thank:

Greg Ferguson, Rob Russell, Terry Davis, Chris Vaill, Jamie Kellogg, Tom Panning, Adam Buckley, Ilia Lobsanov, Giuseppe "Cowo" Corbelli, Florent Fievez, Piero Ottuzzi, Kees Cook, Andrew Newman, David DeVault, Nico Genimakis, Jan−Erik Stromquist.

And, most of all, I'd like to thank my lovely wife, Marta, for her love, support and understanding