Linux MP3 Branden van een CD mini-HOWTO

Greg Wierzchowski

greg.wierzchowski@usa.net

Vertaald door: Bokhorst

bokkie@nl.linux.org

Een compleet recept voor het aanmaken van CD's met MP3 bestanden.

Inhoudsopgave

1. Introductie

Deze mini-HOWTO werd gecreëerd vanwege mijn ervaring met het branden van muziek-CD's en het gebrek aan wat specifieke informatie over geluidsnormalisatie op het Internet. Ik brand gewoonlijk muziek-CD's als een mix - verschillende liedjes vanaf verschillende bronnen. Erg vaak zit er tussen liedjes een enorm verschil in het volumeniveau. Dit is het eerste obstakel. Ten tweede zijn veel van de bestanden op het Internet niet CD-compatibel (16 bit, stereo, 44.1 kHz) en moeten ze worden geconverteerd. Er zijn heel wat programma's om muziek-CD's van MP3 bestanden te branden, en veel daarvan voeren de conversie transparant uit. Maar ik heb nog geen enkel hulpmiddel gezien dat ook het volume normaliseert, dus daarom werkte ik mijn eigen recept over CD-branden uit.

Ik veronderstel dat je een CD wilt branden met de verzameling liedjes die je van verschillende bronnen hebt verkregen, allen variërend in kwaliteit, maar je wilt een CD met de best mogelijke geluidskwaliteit. In deze mini-HOWTO worden de stappen uiteengezet die je daarbij wellicht kunnen helpen.

1.1. Copyright en Licentie

2. Voorbereiden van de tracks

Noot

Alle opdrachten veronderstellen de bashshell

1. Verzamel alle MP3 bestanden in één directory.
2. Als zich daaronder bestandsnamen met spaties bevinden, zet je deze eerst om in onderstrepingstekens:

   ```
   for i in *.mp3; do mv "$i" `echo $i | tr ' ' '_'`; done
   ```

3. Converteer ze naar WAV met de opdracht:

   ```
   for i in *.mp3; do mpg123 -w `basename $i .mp3`.wav $i; done
   ```

   Als het goed is, bevindt Mpg123 zich in elke Linux distributie, maar mocht je het niet hebben, dan kun je het ophalen van http://www.mpg123.de/.

   NOOT Ik bemerkte bij een aantal MP3 bestanden dat de uitvoer van mpg123 werd vervormd. Eerst dacht ik dat de MP3's slecht waren, maar toen ik dit controleerde met een andere speler, klonken ze goed. Dus zocht ik naar een andere MP3 speler welke WAV bestanden naar disk kon schrijven, en vond: MAD mp3 decoder op http://www.mars.org/home/rob/proj/mpeg/. Met madplayer, luidt de opdrachtregel:

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

   Er is nog een andere manier om de conversie uit te voeren. Een aantal MP3 bestanden zorgen zowel met mpg123 als met madplay voor problemen bij het decoderen. De lame encoder, die een decodeer modus heeft, schijnt moeilijke gevallen zeer goed af te handelen (lame is te vinden op http://www.mp3dev.org/mp3/) :

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

   NOOT: De opdracht `basename $i .mp3`.wav vervangt MP3 extensies door WAV. Er zijn 101 manieren om dat te doen, hier is een alternatief: `echo "$1" | sed 's/\.mp3$/.wav/'`

4. Voer de opdracht "file *.wav" uit en controleer de uitvoer op bestanden anders dan 16 bit, stereo 44100 Hz.
5. Mochten er bestanden met verschillende karakteristieken zijn, converteer ze dan naar bovenstaande specs. Om bijvoorbeeld een bestand track01.wav te converteren om een sample rate van 44.1 kHz te verkrijgen, zou je kunnen gebruiken:

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

   Sox is zo populair, dat het waarschijnlijk al standaard is geïnstalleerd bij elke distributie. Het kan worden verkregen vanaf http://www.spies.com/Sox/. De opdrachtregelopties zijn echter wat cryptisch voor de alledaagse gebruiker (ikzelf). Kijk op http://www.spies.com/Sox/sox.tips.html voor wat tips over het gebruik.

6. Normaliseer je WAV bestanden, om drastische verschillen in volumeniveaus te voorkomen. Ik gebruik een programma van Chris Vaill (<cvaill@cs.columbia.edu>), genaamd normalize - het is te verkrijgen vanaf http://www.cs.columbia.edu/~cvaill/normalize/

   Ik gebruik de volgende syntax (-m is voor mix modus, waarbij alle bestanden zo luid mogelijk zouden moeten zijn):

   ```
   normalize -m *.wav
   ```

3. Branden van je CD

Er zijn heel wat programma's voor het maken van CD's met WAV bestanden. Ik gebruik cdrecord voor het branden via de opdrachtregel en XCDROAST tijdens het werken met een gui. Om cdrecord te kunnen gebruiken, moet je weten welk SCSI-device je CD-writer is. Als je een ATAPI writer gebruikt, gebruik je SCSI-emulatie (kernel module ide-scsi). Laten we ervan uitgaan dat je ATAPI cdwriter is aangesloten op de tweede IDE-bus als master. Dus dan heeft het de `/dev/hdc` device file. Om de kernel te instrueren dat je het als een SCSI-device wilt behandelen, voeg je de volgende regel toe aan `/etc/lilo.conf`:

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

Als de kernel de ide-scsi module niet automatisch laadt, dan moet je bovendien insmod ide-scsi in het bestand `rc.local` (of equivalente bestand) plaatsen. Zodra je CD-writer wordt herkend als een SCSI-device, start je cdrecord --scanbus om erachter te komen wat de "dev" parameter voor cdrecord is. Op mijn systeem, ziet de uitvoer er ongeveer zo uit:

```
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
```

Dus de opdracht cdrecord bevat dev=1,1,0 om het device aan te geven. Dit is de volledige opdracht voor mijn systeem:

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

NOOT

Het -pad argument is noodzakelijk, omdat alle audio tracks op de CD moeten worden aangepast op de juiste gegevenslengte, wat niet altijd het geval is met mp3 bestanden.

4. Branden van een DAO CD

DAO, Disc-At-Once, is op het moment de enige methode om een CD te branden zonder een 2 seconden durende pauze tussen de tracks. Het is handig voor het branden van party mixen. Het programma voor het branden van CD's in DAO modus is cdrdao, beschikbaar vanaf SourceForge, http://sourceforge.net/projects/cdrdao/.

Het programma cdrdao maakt gebruik van beschrijvende bestanden genaamd TOC (Table Of Contents=inhoudsopgave, natuurlijk). Er zijn twee manieren om een dergelijk bestand te schrijven. Ten eerste met behulp van een shellscript, welke wordt gedistribueerd met de cdrdao broncode (in de directory `contrib`, genaamd generate_toc.sh. Het accepteert als argument een lijst met `.wav` bestanden en produceert een `cd.toc` bestand. De tweede manier is eenvoudigweg zelf een dergelijk bestand te schrijven in een teksteditor naar keuze. Hier is een voor zichzelf sprekend voorbeeld:

```
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
```

De 0 (nul) achter de wave bestandsnaam betekent: start vanaf het begin van het bestand. Er kan nog een tweede nummer worden opgegeven dat de lengte (in tijd) van het bestand naar het record aangeeft. De xcdroast maakt vergelijkbare TOC bestanden aan. Er bevinden zich ook voorbeelden in de directory `testtocs` van de broncode van cdrdao.

De cdrdao gebruikt standaard het device `/dev/cdrecorder`, wat een link moet zijn naar het cdwriter device. Maak als volgt (als root) de link aan, ervan uitgaande dat je cd recorder device het bestand `/dev/scd0` is:

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

Dan in de veronderstelling dat het TOC bestand de naam `cd.toc` heeft, is de opdracht om de cd te branden eenvoudigweg:

```
cdrdao write cd.toc
```

5. Krediet

Speciale dank gaat uit naar al die mensen die een bijdrage aan de Linux gemeenschap leverden en die deze HOWTO mogelijk maakten.

5.1. Vertalingen

- Im Eunjea - Vertaalde dit document naar Koreaans, URL is http://kltp.kldp.org/eunjea/mp3_burning/.

- Mendel L Chan - Vertaalde dit document naar de Chinese taal, URL is http://www.linux.org.tw/CLDP/mini/MP3-CD-Burning/.

5.2. Overige krediet

- Greg Ferguson - Converteerde dit document in eerste instantie van HTML naar SGML.
- Rob Russell - Corrigeerde mijn naamconversievoorbeeld.
- Terry Davis - Deed de suggestie mijn HOWTO aan te leveren bij linuxdoc.
- Chris Vaill - Maakte het normalize programma.
- Jamie Kellogg - Kwam als oplossing met lame om probleembestanden te decoderen.