

Compilation de VLC 0.8.6d pour streaming vidéo en h264 pour le multiposte de la Freebox V4 - Ubuntu Studio 8.04

COLLABORATORS

	<i>TITLE :</i> Compilation de VLC 0.8.6d pour streaming vidéo en h264 pour le multiposte de la Freebox V4 - Ubuntu Studio 8.04	<i>REFERENCE :</i>	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Jean David TECHER	2009-03-07	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Motivations	1
2	Ce dont on a besoin	1
3	Mise à jour des variables	2
4	Faac	2
5	Lame	2
6	Ffmpeg	3
7	Dvbpsi	3
8	Live555	3
9	Mpeg2dec	3
10	x264	3
11	A52	4
12	VLC	4
13	Script shell pour le multiposte	5

List of Figures

1	Configuration pour la réception du Flux de la FreeBox	1
---	---	---

Abstract

Le présent document détaille les compilations que j'ai effectué pour pouvoir faire du streaming pour le multiposte de Free.

1 Motivations

Cet article n'a rien d'original dans la mesure où je reprends ici les notes que j'ai trouvé pour compiler cette version de VLC. J'ai surtout repris les notes publiées pour l'un des administrateurs unreal du site slashorg.net. J'ai juste ajouté parfois mes propres options de compilation notamment pour VLC.

Actuellement ma configuration pour la FreeBox est la suivante. Mon but est de pouvoir bénéficier de la réception du flux multiposte de ma FreeBox à Montpellier (chez un ami de Montpellier) vers mon appartement actuel sur Antibes...

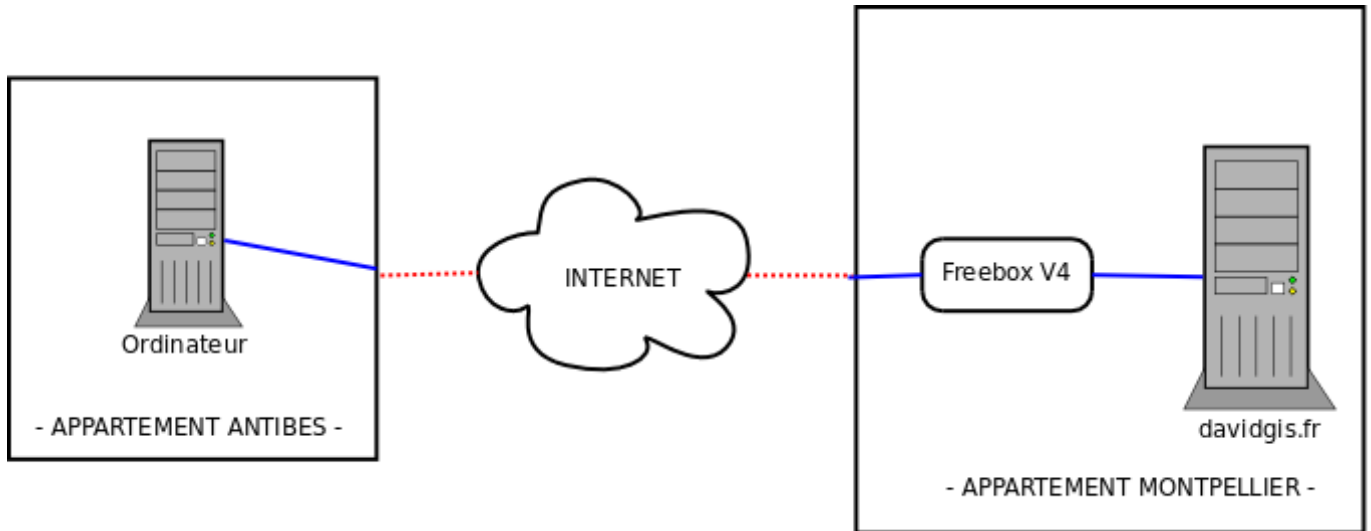


Figure 1: Configuration pour la réception du Flux de la FreeBox

. Plusieurs inconvénients sont déjà présents

- Le souci pour la version présente dans Ubuntu 8.04/8.10 provient du fait que l'option pour transcoder en `acodec=mp4a` se solde par un échec. On aura donc la vidéo mais pas le son. On peut tout aussi bien préciser l'option `acodec=mpga` mais cela ne résout pas les problèmes suivants
- Possédant un appareil photo numérique TYPE MINOLTA que je transfère parfois sur mon serveur, il arrive aussi que la version de VLC 0.8.6 livrée avec les paquets Ubuntu ne prennent pas en compte les fichiers .MOV pour en faire du streaming
- Mon serveur possède déjà une version de VLC que je n'ai pas envie de supprimer. Je la laisse pour l'occasion.
- La version estampillée 0.9.8a de VLC crash quand on applique une playlist en demandant un transcodage du flux en h264.

La solution est donc de revenir à une ancienne version de VLC et de recompiler VLC avec les bibliothèques nécessaires sans avoir à les installer. Le mieux sera donc de compiler VLC en statique. Je propose donc ici un processus de compilation qui permette de parer aux problèmes cités précédemment.

Note

Pour mes compilations, je me suis servi à 80% du travail de unreal à <http://www.slashorg.net/read-100-Tuto-diffusion-video-avec-VLC.html>

2 Ce dont on a besoin

Il faudra les bibliothèques/outils suivants

1. faac-1.26.tar.gz
2. lame-3.97.tar.gz
3. ffmpeg notamment la révision 11377 du depot SVN
4. libdvbpsi5-0.1.6.tar.gz
5. live555-latest.tar.gz
6. mpeg2dec-0.4.1.tar.gz
7. x264-snapshot-20080102-2245.tar.bz2
8. a52dec-0.7.4.tar.gz
9. vlc-0.8.6d.tar.bz2

On va donc installer les différents outils en procédant comme suit



Warning

De 1 à 8, les outils seront à compiler mais ne requiert pas d'installation donc PAS DE MAKE INSTALL. Les librairies sont compilées de manière statique (sauf les librairies systèmes).

3 Mise à jour des variables

Je déposerais l'intégralité de mes sources dans le répertoire SRC définit comme suit

```
export SRC="/mnt/vlc-src"  
export CFLAGS="-I$SRC/faac/include -I$SRC/lame/include"  
export CXXFLAGS=$CFLAGS  
export LDFLAGS="-L$SRC/faac/libfaac/.libs -L$SRC/lame/libmp3lame/.libs"
```

4 Faac

On télécharge et on compile en faisant

```
cd $SRC;  
wget http://surfnnet.dl.sourceforge.net/sourceforge/faac/faac-1.26.tar.gz -O - | tar xzf -  
cd faac  
./bootstrap && ./configure --enable-static --enable-shared=no && make
```

5 Lame

Ici on ruse en prenant la version 3.97 au lieu de la 3.98

```
cd $SRC;  
wget http://freefr.dl.sourceforge.net/sourceforge/lame/lame-3.97.tar.gz -O - | tar xzf -  
mv lame-3.97 lame  
cd lame  
./configure --enable-static=yes --enable-shared=no && make
```

6 Ffmpeg

En prenant en compte les notes de unreal et en remontant les logs de SVN de ffmpeg, il a compilé ses sources de ffmeq pour la snapshot du 2008-01-02. Ce qui correspond à la révision 11377. On va donc commencer par récupérer les sources associés à cette révision

```
cd $SRC;  
svn checkout --revision 11377 svn://svn.mplayerhq.hu/ffmpeg/trunk ffmpeg
```

Conformément à ses options, on va procéder comme lui en ajoutant l'option **--enable-libmp3lame** une fois l'adaption fournie

```
./configure --enable-gpl --enable-pp --disable-shared --enable-static --disable-debug \  
--enable-libmp3lame --enable-libfaac --enable-pthreads \  
--extra-ldflags="-L$SRC/faac/libfaac/.libs -L$SRC/lame/libmp3lame/.libs" \  
--extra-cflags="-I$SRC/faac/include -I$SRC/lame/include" && make
```

7 Dvbpsi

On procède donc ainsi

```
cd $SRC  
wget ftp://ftp.linuxforum.hu/mirrors/frugalware/pub/frugalware/frugalware-testing/source/ ←  
multimedia/libdvbpsi/libdvbpsi5-0.1.6.tar.gz -O - | tar xzf -  
mv libdvbpsi5-0.1.6 libdvbpsi  
cd libdvbpsi  
./bootstrap && ./configure --enable-static --enable-shared=no --enable-release && make
```

8 Live555

Ici on effectue les commandes suivantes

```
cd $SRC  
wget http://www.live555.com/liveMedia/public/live555-latest.tar.gz -O - | tar xzf -  
cd live && ./genMakefiles linux && make
```

9 Mpeg2dec

Une compilation usuelle là aussi

```
cd $SRC  
wget http://libmpeg2.sourceforge.net/files/mpeg2dec-0.4.1.tar.gz -O - | tar xzf -  
mv mpeg2dec-0.4.1 mpeg2dec  
cd mpeg2dec  
./configure --enable-static --enable-shared=no && make
```

10 x264

Une des librairies les plus importantes

```
cd $SRC  
wget ftp://ftp.videolan.org/pub/videolan/x264/snapshots/x264-snapshot-20080102-2245.tar.bz2 ←  
-O - | tar xjf -  
mv x264-snapshot-20080102-2245 x264  
cd x264 && ./configure --enable-pthread && make
```

11 A52

On fait

```
cd $SRC
wget http://liba52.sourceforge.net/files/a52dec-0.7.4.tar.gz -O - | tar xzf -
mv a52dec-0.7.4 a52
cd a52
./configure --enable-static=yes --enable-shared=no && make
```

Il est temps maintenant de compiler VLC :)

12 VLC

Ici les options sont plus importantes comme toujours avec VLC

```
cd $SRC

wget http://downloads.videolan.org/pub/videolan/vlc/0.8.6d/vlc-0.8.6d.tar.bz2 -O - | tar ←
  xjf -

mv vlc-0.8.6d vlc
cd vlc

./configure --enable-release --enable-static --enable-sout --enable-live555 \
--enable-ffmpeg --with-ffmpeg-faac --with-ffmpeg-mp3lame --enable-libmpeg2 \
--enable-x264 --enable-ncurses --enable-dvbpsi --with-included-gettext \
--enable-a52 --disable-x11 --disable-mad --disable-growl --disable-notify \
--disable-dvdnav --disable-gnomevfs --disable-libcdio --disable-libcddb \
--disable-cdda --disable-vcd --disable-screen --disable-ogg --disable-mkv \
--disable-mod --disable-mpc --disable-quicktime --disable-dts \
--disable-vorbis --disable-speex --disable-png --disable-cmml \
--disable-xvideo --disable-glx --disable-xinerama --disable-opengl \
--disable-sdl --disable-sdl-image --disable-freetype --disable-fribidi \
--disable-libxml2 --disable-fb --disable-wxwidgets --disable-visual \
--disable-daap --disable-bonjour --disable-gnutls --disable-skins2 \
--with-live555-tree=$SRC/live --with-ffmpeg-tree=$SRC/ffmpeg \
--with-libmpeg2-tree=$SRC/mpeg2dec --with-x264-tree=$SRC/x264 \
--with-dvbpsi-tree=$SRC/libdvbpsi --with-a52-tree=$SRC/a52
#
# On évite l'erreur de compilation remontée par unreal sur mux.c
#
rm -f modules/codec/ffmpeg/mux.c
wget http://www.davidgis.fr/download/mux.c -O modules/codec/ffmpeg/mux.c
make
```

Une fois la compilation terminée, il suffit de récupérer le binaire VLC et de vérifier les dépendances. Chez moi au niveau des dépendances tout est ok pour les codecs qui sont directement dans le binaire... Ensuite je l'ai copié vers /opt/vlc-0.8.6d/bin/ - répertoire que j'ai créé -.

```
root@olivia:/mnt/vlc-src/vlc# ldd vlc
linux-gate.so.1 => (0xffffe000)
libhal.so.1 => /usr/lib/libhal.so.1 (0xb7ec8000)
libdbus-1.so.3 => /usr/lib/libdbus-1.so.3 (0xb7e93000)
librt.so.1 => /lib/tls/i686/cmox/librt.so.1 (0xb7e89000)
libpthread.so.0 => /lib/tls/i686/cmox/libpthread.so.0 (0xb7e71000)
libdl.so.2 => /lib/tls/i686/cmox/libdl.so.2 (0xb7e6d000)
libz.so.1 => /usr/lib/libz.so.1 (0xb7e58000)
libstdc++.so.6 => /usr/lib/libstdc++.so.6 (0xb7d65000)
```



```
libm.so.6 => /lib/tls/i686/cmov/libm.so.6 (0xb7d40000)
libgcc_s.so.1 => /lib/libgcc_s.so.1 (0xb7d35000)
libc.so.6 => /lib/tls/i686/cmov/libc.so.6 (0xb7bea000)
/lib/ld-linux.so.2 (0xb7ef4000)
```

13 Script shell pour le multiposte

On pourra par exemple proposer le script Bash suivant

```
MULTIPOSTE_PLAYLIST="playlist.m3u";
# On telecharge la playlist...
[ ! -f "${MULTIPOSTE_PLAYLIST}" ] && rm -f "${MULTIPOSTE_PLAYLIST}" ;
echo "info:mise a jour de la playlist: ${MULTIPOSTE_PLAYLIST}";
wget -q 'http://mafreebox.freebox.fr/freeboxtv/playlist.m3u' -O "${MULTIPOSTE_PLAYLIST}";
vlc "${MULTIPOSTE_PLAYLIST}" --sout '#transcode{vcodec=h264,vb=256,acodec=mp4a,ab=32,↵
channels=1,samplerate=48000,width=320,height=240,deinterlace}:standard{mux=asf,dst↵
=:8090,access=http}:sout-transcode-soverlay=0' -I ncurses 2> /dev/null;
exit 0;
```