VideoLAN HOWTO

Alexis de Lattre

Marc Ariberti

Anil Daoud

Clément Stenac

v2.0, 2003−09−04

This document describes how to use the complete VideoLAN streaming solution.

Table of Contents

Table of Contents

Table of Contents

Chapter 1. Introduction

1.1. What is the VideoLAN project ?

1.1.1. Overview

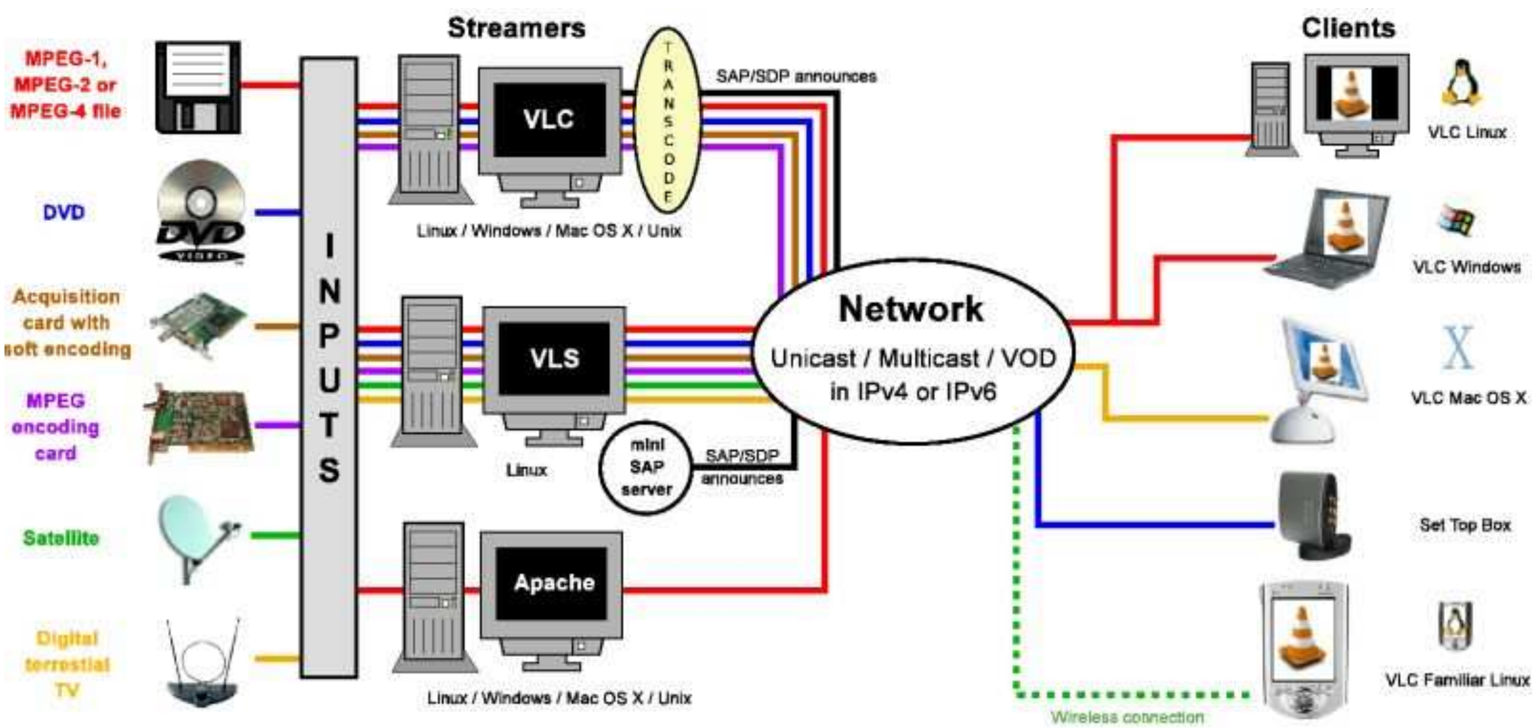
VideoLAN is a complete software solution for video streaming, developed by students of the Ecole Centrale Paris and developers from all over the world, under the GNU General Public License (GPL). VideoLAN is designed to stream MPEG videos on high bandwidth networks.

The VideoLAN solution includes :

- VLS (VideoLAN Server), which can stream MPEG−1, MPEG−2 and MPEG−4 files, DVDs, digital satellite channels, digital terrestial television channels and live videos on the network in unicast or multicast,
- VLC (initially VideoLAN Client), which can be used as a server to stream MPEG−1, MPEG−2 and MPEG−4 files, DVDs and live videos on the network in unicast or multicast ; or used as a client to receive, decode and display MPEG streams under multiple operating systems.

Here is an illustration of the complete VideoLAN solution :

Figure 1−1. Global VideoLAN solution La solution VideoLAN globale



More details about the project can be found on the VideoLAN Web site.

1.1.2. VideoLAN software

1.1.2.1. VLC

VLC works on many platforms : Linux, Windows, Mac OS X, BeOS, *BSD, Solaris, Familiar Linux, Yopy/Linupy and QNX. It can read :

- MPEG−1, MPEG−2 and MPEG−4 / DivX files from a hard disk, a CD−ROM drive, ...
- DVDs and VCDs,
- from a satellite card (DVB−S),
- MPEG−1, MPEG−2 and MPEG−4 streams from the network sent by VLS or VLC's stream output.

VLC can also be used as a server to stream :

- MPEG−1, MPEG−2 and MPEG−4 / DivX files,
- DVDs,
- from an MPEG encoding card,

to :

- one machine (i.e. to one IP address) : this is called *unicast*,
- a dynamic group of machines that the clients can join or leave (i.e. to a multicast IP address) : this is called *multicast*,

in IPv4 or IPv6 .

To get the complete list of VLC's possibilities on each plateform supported, see the VLC features page.

VLC doesn't work on Mac OS 9, and will probably never do.

1.1.2.2. VLS

VLS can stream :

- an MPEG−1, MPEG−2 or MPEG−4 files stored on a hard drive or on a CD,
- a DVD located in a local DVD drive or copied on a hard disk,
- a satellite card (DVB−S) or a digital terrestial television card (DVB−T) ,
- an MPEG encoding card ;

to:

- one machine (i.e. to one IP address) : this is called *unicast*,
- a dynamic group of machines that the clients can join or leave (i.e. to a multicast IP address) : this is called *multicast*,

in IPv4 or IPv6 .

A Pentium 100 MHz with 32 MB of memory should be enough to send one stream on the network. When streaming a lot of videos stored on a hard drive, the actual limitation is not the processor but the hard drive and the network connection.

VLS works under Linux and Windows. To get the complete list of VLS's possibilities on each plateform supported, see the streaming features page.

1.1.2.3. Mini−SAP−server

You can add a channel information service based on the SAP/SDP standard to the VideoLAN solution. The mini−SAP−server sends announces about the multicast programs on the network in IPv4 or IPv6, and VLCs receive these annouces and automatically add the programs announced to their playlist.

The mini−SAP−server works under Linux and Mac OS X.

1.2. What is a codec ?

To fully understand the VideoLAN solution, you must understand the difference between a *codec* and a *container format*

- A *codec* is a compression algorithm, used to reduce the size of a stream. There are audio codecs and video codecs. MPEG−1, MPEG−2, MPEG−4, Vorbis, DivX, ... are codecs
- A *container format* contains one or several streams already encoded by codecs. Very often, there is an audio stream and a video one. AVI, Ogg, MOV, ASF, ... are container formats. The streams contained can be encoded using different codecs. In a perfect world, you could put any codec in any container format. Unfortunately, there are some incompatibilities. You can find a matrix of possible codecs and container formats on the features page

To decode a stream, VLC first *demuxes* it. This means that it reads the container format and separates audio, video, and subtitles, if any. Then, each of these are passed *decoders* that do the mathematical processing to decompress the streams .

There is a particular thing about MPEG:

- MPEG is a *codec*. There are several versions of it, called MPEG−1, MPEG−2, MPEG−4, ...
- MPEG is also a container format, sometimes refered to as *MPEG System*. There are several types of MPEG: ES, PS, and TS

  When you play an MPEG video from a DVD, for instance, the MPEG stream is actually composed of several streams (called Elementary Streams, ES): there is one stream for video, one for audio, another for subtitles, and so on. These different streams are mixed together into a single Program Stream (PS). So, the .VOB files you can find in a DVD are actually MPEG−PS files. But this PS format is not adapted for streaming video through a network or by satellite, for instance. So, another format called Transport Stream (TS) was designed for streaming MPEG videos through such channels.

1.3. How can I use VideoLAN ?

1.3.1. Documentation

The user documentation of VideoLAN is made up of 4 documents :

- the *VideoLAN Quickstart*. This document will give you a quick overview of of VLC, VLC's stream output, the Video On Demand solution and the channel information service system.

- the *VideoLAN HOWTO*. This document is the complete guide of the VideoLAN streaming solution.
- the *VLC user guide*. This document is the complete guide for VLC.
- the *VLS user guide*. This document is the complete guide for VLS.
- the *VideoLAN FAQ*. This document contains Frequently Asked Questions about VideoLAN.

The latest version of these documents can be found on the documentation page .

You can also have a look at the VideoLAN Wiki. This is a website that everyone can change. We use it to document everything that is not in the "official" documentation: the tips and tricks for each O.S., the graphical interfaces, etc...

1.3.2. User support

If you have problems using VideoLAN, and if you don't find the answer to your problems in the documentation, please look at the online archive of the mailing−lists. There are two English−speaking mailing−lists for the users :

- *vlc@videolan.org* for the questions on VLC ,
- *streaming@videolan.org* for the questions on VLS, mini−SAP−server and the network .

If you want to subscribe or unsubscribe to the mailing−lists, please go to the mailing−list page.

You can also talk with VideoLAN users and developers on IRC : server *irc.freenode.net*, channel *#videolan* .

If you find a bug, please follow the instructions on the bug reporting page .

1.4. Command line usage

- VLC has many different graphical interfaces, that are organized quite differently in order to be in harmony with the guidelines of each operating system supported. Documenting the use of each graphical interface is too long, and some features are only available via the command line interface. Therefore we decided to document only the command line interface, but in many cases it shoud be easy to guess how to use the graphical interface for the same use !
- VLS has a command line and a telnet interface, but no graphical interface !

All the commands that show up in this document should be typed inside a terminal. .

1.4.1. Open a terminal

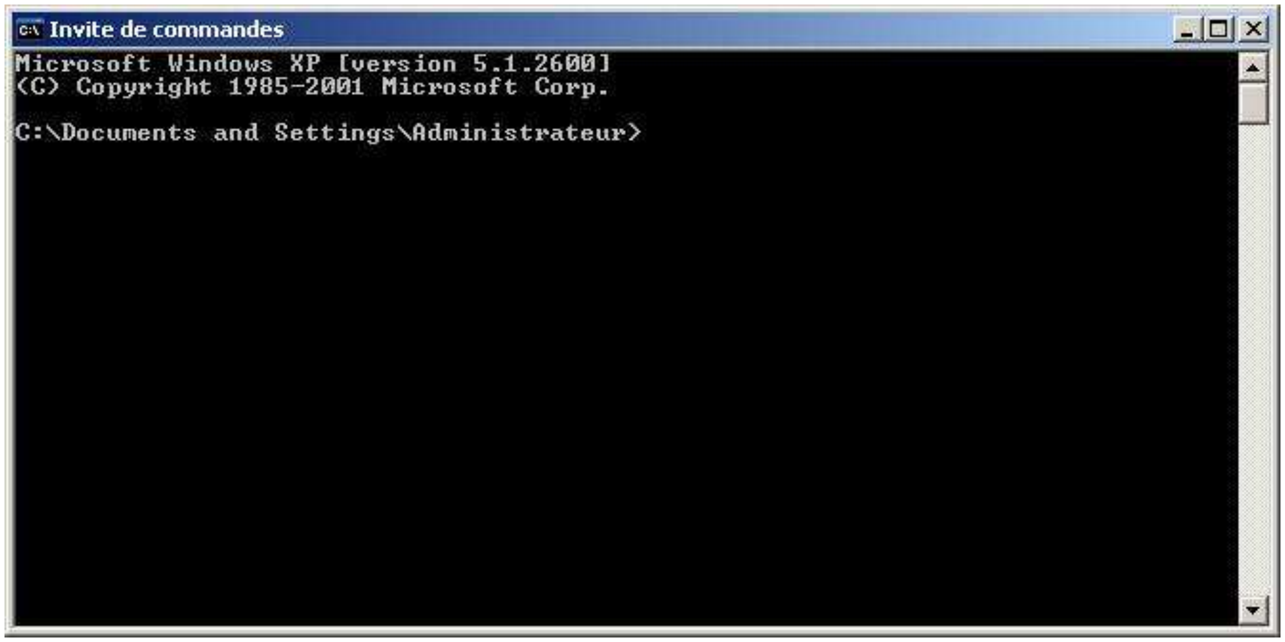
1.4.1.1. Windows

Click on *Start*, *Run* and type :

- **cmd Enter** (Windows 2000 / XP),
- **command Enter** (Windows 95 / 98 / ME).

The terminal appears Le terminal apparait

Figure 1−2. Windows terminal



Under Windows, you need to be in the directory where the program is installed to run it.

1.4.1.2. Linux / Unix

Open a terminal :

Figure 1−3. Linux X terminal

In the documentation, we adopt the following conventions for the Unix commands :

- commands that should be typed as *root* have a **#** prompt :

```
# command_to_be_typed_as_root
```

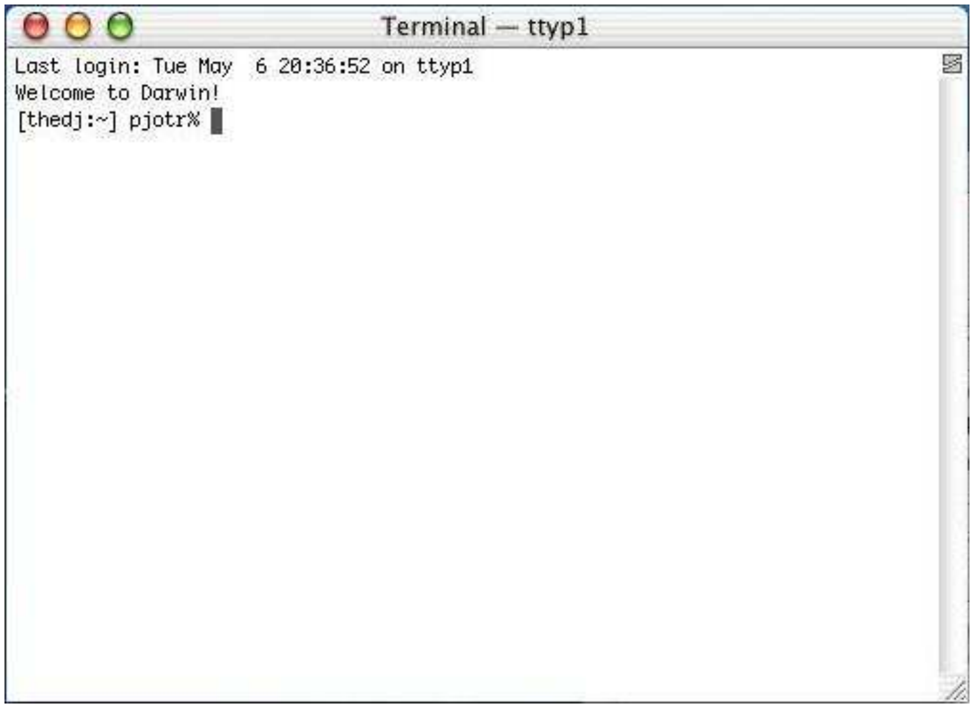
- commands that should be typed as a regular user have a **%** prompt :

```
% command_to_be_typed_as_regular_user
```

1.4.1.3. Mac OS X

Go to *Applications*, open the folder *Utilities* and double−click on *Terminal* :

Figure 1−4. Mac OS X terminal

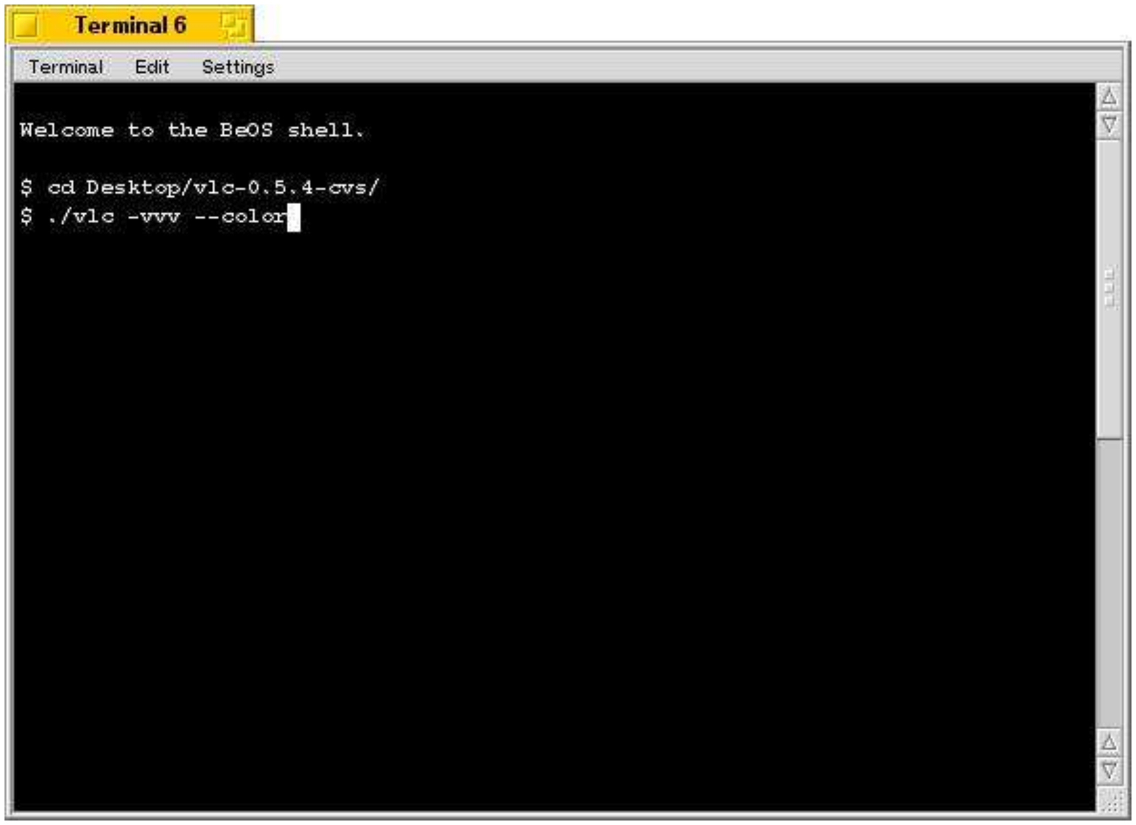


Under Mac OS X, you need to be in the directory where the program is installed to run it, and start the command with **./** .

1.4.1.4. BeOS

In the deskbar, go to *Application* and then *Terminal* :

Figure 1−5. BeOS terminal



Under BeOS, you need to be in the directory where the program is installed to run it, and start the command with **./** .

Chapter 2. Install the required software and hardware

2.1. Install VLC

There are VLC binaries available for the many OSes, but not for all supported OSes. If there are no binaries for your OS or if you want to change the default settings, you can compile VLC from sources.

2.1.1. Windows

VLC works under Windows 95/98/ME/2000/XP. Download the self−extracting file from the VLC Windows download page. Launch the `.exe` to install VLC.

2.1.2. BeOS

Download the Zip file from the VLC BeOS download page. Unzip the file in a directory to install VLC.

2.1.3. Mac OS X

Download the Mac OS X package from the VLC MacOS X download page . Double−click on the icon of the package : an icon will appear on your Desktop, right beside your drive(s). Open it and drag the VLC application from the resulting window to the place where you want to install it (it should be `/Applications`).

2.1.4. Debian GNU/Linux

2.1.4.1. Debian stable (woody)

Add the following lines to your `/etc/apt/sources.list`:

```
deb http://www.videolan.org/pub/videolan/debian $(ARCH)/
deb-src http://www.videolan.org/pub/videolan/debian sources/
```

Then, for a normal install, do:

```
# apt-get update
# apt-get install gnome-vlc libdvdcss2
```

2.1.4.2. Debian unstable (sid)

Add the following lines to your `/etc/apt/sources.list`:

```
deb http://www.videolan.org/pub/videolan/debian $(ARCH)/
deb-src http://www.videolan.org/pub/videolan/debian sources/
```

Then, for a normal install, do:

```
# apt-get update
# apt-get install wxvlc libdvdcss2
```

2.1.4.3. Debian testing (sarge)

You should not be using Debian testing unless you perfectly know what you are doing. It is almost impossible to support Debian testing and there are no plans to do it. For more informations on Debian testing, please look: testing page

2.1.5. Linux Mandrake

There are VLC packages for Mandrake 9.1 and cooker.

To install them, add the following sources for either Mandrake 9.1 or Cooker (you can use Easy urpmi for that): *contrib* from the core distribution and *plf* (Penguin Liberation Front) from the external add−ons.

Then install the required packages with **urpmi**:

```
# urpmi libdvdcss2 libdvdplay0 wxvlc vlc-plugin-a52 vlc-plugin-ogg vlc-plugin-mad
```

2.1.6. Linux Redhat

Download the RPM package *vlc* and the packages listed in the *required libraries and codecs* section (the other packages are just optional) from the VLC Red Hat download page and put them all into the same directory.

Then install the RPM packages you have downloaded:

```
# rpm -U *.rpm
```

If you have not installed all the RPM packages included with your distribution, you may be asked to install a few of them first.

2.1.7. Compile the sources by yourself (for every other OS)

The method below is for any Unix system supported by VLC, for which there is no packages available. It explains how to compile and install VLC and the needed libraries from their source code.

You can also compile VLC under Linux this way if you want to modify the default supported modules.

2.1.7.1. Install the libraries

Many libraries are needed :

- *libdvbpsi* (compulsory) ,
- *mpeg2dec* (compulsory) ,
- *libdvdcss* if you want to be able to read encrypted DVDs ,
- *libdvdplay* if you want to have DVD menu navigation ,
- *a52dec* if you want to be able to decode the AC3 (i.e. A52) sound format often used in DVDs ,
- *ffmpeg*, *libmad*, *faad2* if you want to read MPEG 4 / DivX files ,

- *libogg* & *libvorbis* if you want to read Ogg Vorbis files .

Download the libraries from the VLC sources download page.

For each library :

- uncompress :
  ```
  % tar xvzf library.tar.gz
  ```
  or
  ```
  % tar xvjf library.tar.bz2
  ```
- configure :
  ```
  % cd library
  % ./configure
  ```
- compile and install :
  ```
  % make
  # make install
  ```

Check that the configuration file `/etc/ld.so.conf` contains the following line :

```
/usr/local/lib
```

If the line is not present, add–it and then run (as root):

```
# ldconfig
```

2.1.7.2. Install VLC

Download the sources of the lastest release : get the file `vlc−version.tar.gz` from the VLC sources download page. Uncompress–it :

```
% tar xvzf vlc−version.tar.gz
% cd vlc−version
```

To get the list of configuration options, do :

```
% ./configure −−help
```

Please note that all the modules are described in the *Modules* section of the VLC User Guide .

Examples of very simple configurations:

- if you want a basic VLC, do :
  ```
  % ./configure
  ```
- if you want the Gnome interface instead of the GTK interface (you will need the developement packages of Gnome) :
  ```
  % ./configure −−enable−gnome
  ```

Then, compile and install :

```
% make
```

```
% su
Password:  [Root Password]
# make install
```

Please note that the installation (**make install** command) is not mandatory. You can execute VLC from where you compiled it.

2.2. Install VLS

2.2.1. Windows

Download the ZIP file from the VLS Windows download page, unzip−it and run `setup.exe` .

2.2.2. GNULinux & Mac OS X

2.2.2.1. Install the libraries

Many libraries are needed for particular uses

- *libdvbpsi* (always needed)
- *libdvdcss* if you want to be able to access encrypted DVDs ,
- *libdvdread* if you want to be able to stream DVDs ,
- *libdvb* if you want to be able to stream from a DVB card (a satellite card or a digital terrestial TV card) .

Download the libraries from the VLS sources download page .

For each library, uncompress, configure (unless for *libdvb* which doesn't have a *./configure*), compile and install :

```
% tar xvzf library.tar.gz
% cd library
% ./configure
% make
# make install
```

Check that the configuration file `/etc/ld.so.conf` contains the following line :

```
/usr/local/lib
```

If the line is not present, add−it and then run :

```
# ldconfig
```

2.2.2.2. Install VLS

Download the sources of the latest release : get the file `vls−version.tar.gz` from the VLS sources download page. Uncompress−it and generate *./configure* :

```
% tar xvzf vls−version.tar.gz
```

```
% cd vls-version
```

To get the list of configuration options, do

```
% ./configure --help
```

Then configure vls :

- if you want a basic VLS without DVD support, do :
  ```
  % ./configure --disable-dvd
  ```
- if you want a VLS with DVD support, do :
  ```
  % ./configure
  ```
- if you want a VLS with DVB support, do :
  ```
  % ./configure --enable-dvb --with-dvb=PATH_TO_DVB_DRIVERS --with-libdvb=PATH_TO_LIBDVB
  ```

Then, compile and install :

```
% make
# make install
```

You can also do a **make uninstall**, **make clean** or **make distclean** as needed .

Chapter 3. Receive and save a stream

3.1. Receive a stream with VLC

3.1.1. Receive an unicast stream

```
% vlc -vvv udp:
```

3.1.2. Receive a multicast stream

```
% vlc -vvv udp:@239.255.12.42
```

where *239.255.12.42* is the multicast IP address you want to join .

3.1.3. Receive an HTTP/FTP/MMS stream

Use one of the following command lines :

- `% vlc -vvv http://example/stream.xyz`
  where *http://example/stream.xyz* is the HTTP address of the stream ;
- `% vlc -vvv ftp://example/stream.xyz`
  where *ftp://example/stream.xyz* is the FTP address of the stream ;
- `% vlc -vvv mms://viptvr.yacast.fr/encoderfranceinfo`
  where *mms://viptvr.yacast.fr/encoderfranceinfo* is the MMS address of the stream .

3.2. Save a stream with VLC

VLC can save the stream to the disk. In order to do this, use the Stream Output of VLC : you can do it via the graphical interface, or you can add to the command line the following argument :

```
--sout file/muxer:stream.xyz
```

where :

- *muxer* is one of the formats supported by VLC's stream output, i.e. :
  - ♦ *avi* for AVI format ,
  - ♦ *ogg* for OGG format ,
  - ♦ *ps* for MPEG2−PS format ,
  - ♦ *ts* or *ts_dvbpsi* for MPEG2−PS format (the latter uses *libdvbpsi* for the task) .
- and `stream.xyz` is the name of the file you want to save the stream to, with the right extension .

3.3. Receive a stream with a set−top−box

Some set−top−boxes with Ethernet cards can receive MPEG2−TS streams over UDP and support multicast. This is the case of the Pace and Aminocom set top boxes. We have tested the *AmiNET 100* set−top−box from Aminocom: it is nice and small, and it works very well with VideoLAN .

Chapter 4. Stream a file

4.1. Stream a file with VLC

```
% vlc -vvv video1.xyz --sout udp:192.168.0.42 --ttl 12
```

where :

- `video1.xyz` is the file you want to stream ,
- *192.168.0.42* is either :
    - the IP address of the machine you want to unicast to ;
    - or the DNS name the machine you want to unicast to ;
    - or a multicast IP address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

If you want to stream the file continuously, add the --*loop* option .

4.2. Stream a file with VLS

```
% vls -vv -d udp:192.168.0.42 file:video1.xyz --ttl 12
```

where :

- `video1.xyz` is the file you want to stream ,
- *192.168.0.42* is either :
    - the IP address of the machine you want to unicast to ;
    - or the DNS name the machine you want to unicast to ;
    - or a multicast IP address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

When you want to stop VLS, use the key combination **Ctrl−c**.

If you want to stream the file continuously, add the --*loop* option .

VLS can stream MPEG files that meet two critera :

- the file must be MPEG *PS* (Program Stream) or MPEG *TS* (Transport Stream), that contain video and audio multiplexed. VLS cannot stream MPEG *ES* (Elementary Stream), i.e. a file with only audio or video .

  In order to know if an MPEG file is MPEG *PS*, MPEG *TS* or MPEG *ES*, read the file with VLC and look at the messages (select in the menu *View / Messages*, or use the command line **vlc −vvv**) .

    - If you see a line :
      ```
      [00000107] main module debug: using demux module "ts_dvbpsi"
      ```

it means the file is MPEG TS .

- ♦ If you see a line :

```
[00000109] main module debug: using demux module "ps"
```

it means the file is MPEG PS .

- ♦ If you see a line :

```
[00000109] main module debug: using demux module "es"
```

it means the file is MPEG ES, VLS can't stream it. .

- the sequence header of the video must repeat itself regularly, which is often the case with MPEG−2, but very rare with MPEG−1. There is no easy way to know if the sequence header is repeated regularly. Files with a *.vob* extension are normally MPEG−2 files and files with *.mpg* or *.mpeg* extension or usually MPEG−1 files .

You can download this streamable MPEG−2 PS file for your tests : presentation_short.vob .

Chapter 5. Stream a DVD

Under Unix/Linux, you must have write access to the device corresponding to your DVD drive. For that, you should be in the *disk* or *cdrom* group (look at the permissions in `/dev`). If you're not, add yourself to the group :

```
# adduser your_login disk_or_cdrom
```

and then restart your session .

5.1. Stream a DVD with VLC

```
% vlc -vvv dvdold:/dev/dvd --sout udp:192.168.0.12 --ttl 12
```

where :

- `/dev/dvd` is the name of your DVD drive (put `D:` under Windows if `D` is the letter of your DVD drive) or the directory where you copied your DVD ,
- *192.168.0.42* is either :
  - the IP address of the machine you want to unicast to ;
  - or the DNS name the machine you want to unicast to ;
  - or a multicast IP address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

If you want to stream the DVD continuously, add the *--loop* option .

5.2. Stream a DVD with VLS

This is possible under GNU/Linux only .

```
% vls -vv -d udp:192.168.0.42 dvd:/dev/dvd --ttl 12
```

where :

- `/dev/dvd` is the name of your DVD drive or the directory where you copied your DVD ,
- *192.168.0.42* is either :
  - the IP address of the machine you want to unicast to ;
  - or the DNS name the machine you want to unicast to ;
  - or a multicast IP address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

If you want to stream the DVD continuously, add the *--loop* option .

Chapter 6. Stream a DVB channel (satellite or digital terrestial TV)

This is possible under GNU/Linux only .

6.1. Install the DVB drivers

If you want to be able to stream from a DVB card (a satellite card or a digital terrestial TV card), you need to install the DVB drivers:

- if you use a Linux 2.6.x kernel, you just need to compile the right modules ;
- if you are using a Linux 2.4.x kernel, you must download the latest release of the DVB drivers from the DVB drivers download page of the LinuxTV Project.

Uncompress the tarball and follow the instructions written in the `INSTALL` file to compile and install the drivers .

6.2. Stream with VLS

Put a `.dvbrc` file containing the DVB channels (satellite or digital terrestial TV channels) you want to stream in your home directory (some are provided in the *libdvb* tarball for the satellite channels) .

Run VLS with the following command line :

```
% vls -vv -d udp:192.168.0.42 dvb:"EUROSPORT" --ttl 12
```

where :

- *"EUROSPORT"* is the channel you want to stream as written in your `~/.dvbrc` file ,
- *192.168.0.42* is either :
  - the IP address of the machine you want to unicast to ;
  - or the DNS name the machine you want to unicast to ;
  - or a multicast IP address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

6.3. Stream with VLC

For this, you must use the CVS version of VLC and compile it with **./configure --enable-dvb**
This feature should still be considered as experimental.

```
% vlc -vvv --program 8211 dvb:12597000:0:3:27500000 --sout udp:192.168.0.12 --ttl 12
```

where :

* `8211` is the PID of the DVB channel you would like to stream, expressed as a decimal number ;
* `12597000` is the frequency of the transponder you want to stream from in Hz;
* `0` is the polarization (0 for vertical, 1 for horizontal) ;
* `3` is the FEC (3 for 3/4, 9 for auto) ;
* `27500000` is the transponder symbol rate in Hz ;
* *192.168.0.12* is either :
    * the IP address of the machine you want to unicast to ;
    * or the DNS name the machine you want to unicast to ;
    * or the multicast IP address.
* *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers).

Chapter 7. Stream from an MPEG encoding card

This is possible under GNU/Linux only .

VideoLAN supports two MPEG−2 encoding cards :

- Hauppauge WinTV−PVR−250,
- Visiontech Kfir.

The Hauppauge WinTV−PVR−250 gives much better results and is cheaper than the Visiontech Kfir .

7.1. Stream with the Hauppauge WinTV−PVR−250 card

7.1.1. Install the drivers

First, you will have to patch your kernel to support the v4l2 API (Video 4 Linux version 2). The patch is available on the Video4Linux HQ .

Once your kernel is ready, install the CVS version of the Linux drivers for the Hauppauge WinTV−PVR−250. They are hosted on ivtv.sourceforge.net. Then, you will have to create the device and load the modules; for this, please refer to the documentation shipped with the drivers .

7.1.2. Stream with VLC

You must add **−−enable−pvr** to **./configure** to use this feature. You may want to change the value of *CR_MAX_AVERAGE_COUNTER* to *1000* in `src/input/input_clock.c` to work around a problem with the card .

```
% vlc −vvv pvr:/dev/video0:norm=secam:size=720x576:frequency=576250:
bitrate=3000000:maxbitrate=4000000 −−sout udp:192.168.0.42 −−ttl 12
```

where :

- */dev/video0* is the device corresponding to the encoding card ,
- *norm=secam* is name of the standard of the analogic signal (possible values are pal, secam, and ntsc) ,
- *size=720x576* is the size of the video you want to stream ,
- *frequency=567250* is the frequency in kHz of the channel you want to stream ,
- *bitrate=3000000* is the average bitrate of the stream ,
- *maxbitrate=4000000* is the maximum bitrate of the stream ,
- *192.168.0.42* is either :
    - the IP address of the machine you want to unicast to ;
    - or the DNS name the machine you want to unicast to ;
    - or a multicast IP address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

7.1.3. Stream with VLS

There is no command line interface for this input for the moment, so you will have to use the configuration file .

You cannot set−up the parameters of the drivers (norm, size, frequency, bitrate, etc...) with VLS: for that, you will have to use the program **ptune.pl** that come with the drivers. Please refer to the documentation shipped with the drivers to learn how to use this program .

Here is a typical configuration file `vls.cfg`to run VLS with a Hauppauge WinTV−PVR−250 encoding card :

```
# vls.cfg (VLS configuration file)
# Example of the VideoLAN HOWTO for the Hauppauge WinTV-PVR-250 encoding card

BEGIN "Inputs"

  pvr       = "video"

END

BEGIN "pvr"

  # Hauppauge WinTV-PVR-250 device
  Device        = "/dev/video0"

  # Stream type
  Type          = "Mpeg2-PS"

END

BEGIN "Channels"

  channel1       = "network"

END

BEGIN "channel1"

  # Unicast or multicast IP address
  DstHost = "192.168.0.42"

  # Destination port
  DstPort = "1234"

  # If it's a multicast IP address, uncomment the line below
  #Type    = "multicast"

  # If it's a multicast IP address, set the "Time To Live" below
  #TTL     = "12"

END

BEGIN "LaunchOnStartUp"

  command1 = "start video channel1 pvr"

END
```

Once you have adapted the configuration file above, run VLS :

```
% vls -vv -f vls.cfg
```

7.2. Stream with the Visiontech Kfir card

7.2.1. Install the drivers

If you want to be able to stream from a Visiontech Kfir card, you need to install its Linux drivers. Download the latest release of the drivers from the drivers download page of the LinuxTV web site .

Uncompress the tarball and follow the instructions written in the `INSTALL` file to compile and install the drivers .

If you have a VIA chipset, you need to disable USB in the BIOS .

7.2.2. Stream with VLC

```
% vlc -vvv kfir:/dev/video --sout udp:192.168.0.42 --ttl 12
```

where :

- `/dev/video` is the device corresponding to the Kfir card ,
- *192.168.0.42* is either :
    - the IP address of the machine you want to unicast to ;
    - or the DNS name the machine you want to unicast to ;
    - or a multicast IP address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

7.2.3. Stream with VLS

There is no command line interface for this input for the moment, so you will have to use the configuration file .

Here is a typical configuration file `vls.cfg`to run VLS with a VisionTech Kfir encoding card :

```
# vls.cfg (VLS configuration file)
# Example of the VideoLAN HOWTO for the Visiontech Kfir encoding card

BEGIN "Inputs"

  kfir       = "video"

END

BEGIN "kfir"

  # Visiontech Kfir device
  Device        = "/dev/video"
```

```
  # Stream type (default is "Mpeg2-PS")
  Type          = "Mpeg2-PS"

END

BEGIN "Channels"

  channel1       = "network"

END

BEGIN "channel1"

  # Unicast or multicast IP address
  DstHost = "192.168.0.42"

  # Destination port
  DstPort = "1234"

  # If it's a multicast IP address, uncomment the line below
  #Type    = "multicast"

  # If it's a multicast IP address, set the "Time To Live" below
  #TTL     = "12"

END

BEGIN "LaunchOnStartUp"

  command1 = "start video channel1 kfir"

END
```

Once you have adapted the configuration file above, run VLS :

```
% vls -vv -f vls.cfg
```

Chapter 8. Stream from an acquisition card or a webcam

This is possible under GNU/Linux only .

8.1. Install the Video for Linux drivers

If you want to stream from an acquisition card or a webcam, a video4linux driver must be available for it. You can find more information about video4linux and supported devices here .

Compile the right module for your device, and insert it into your kernel (Some video4linux modules are shipped with the 2.4.x Linux kernels). You can test your device by using any of the listed programs in the *Video: TV and PVR/DVR* section of this page .

Note that v4l2 modules will also work with VLC .

8.2. Stream with VLC

You must add **−−enable−v4l** to **./configure** to use this feature .

```
% vlc -vvv v4l:/dev/video:norm=secam:frequency=543250:size=640x480:channel=0:adev=/dev/dsp:audio=0
--sout '#transcode{vcodec=mp4v,acodec=mpga,vb=3000,ab=256,vt=800000,keyint=80,deinterlace}:std{access=udp,mux=ts,url=239.255.12.13}' --ttl 12
```

where :

- `/dev/video` is the device corresponding to your acquisition card or your webcam ,
- *norm=secam* is name of the standard of the analogic signal (possible values are pal, secam, and ntsc) ,
- *frequency=543250* is the frequency of the channel in khz (*Warning :* for VLC < 0.6.1, Frequency is channel frequency in MHz multiplied by 16) ,
- *size=640x480* is the size of the video you want (you can also put the standard size like *subqcif* (128x96), *qsif* (160x120), *qcif* (176x144), *sif* (320x240), *cif* (352x288) or *vga* (640x480)) ,
- *channel=0* is the number of the channel (usually 0 is for tuner, 1 for composite and 2 for svideo) ,
- *adev=/dev/dsp* is the audio device ,
- *audio=1* is the number of the audio channel (usually 0 is for mono and 1 for stereo) ,
- *vcodec=mp4v* is the video format you want to encode in (*mp4v* is MPEG−4, *mpgv* is MPEG−1, and there is also *h263*, *DIV1*, *DIV2*, *DIV3*, *I420*, *I422*, *I444*, *RV24*, *YUY2*),
- *acodec=mpga* is the audio format you want to encode in (*mpga* is MPEG audio layer 2, *a52* is A52 i.e. AC3 sound) ,
- *vb=3000* is the video bitrate in Kbit/s ,
- *ab=256* is the audio bitrate in Kbit/s
- *vt=800000* is the video bitrate tolerance in bit/s ,
- *keyint=80* is the maximum interval between two "I" images in number of images ,
- *deinterlace* tells VLC to deinterlace the video on the fly ,
- *192.168.0.42* is either :
    - ♦ the IP address of the machine you want to unicast to ;
    - ♦ or the DNS name the machine you want to unicast to ;

    ♦ or a multicast IP address .
* *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

8.3. Stream with VLS

You need the CVS version of VLS. When you compile it, you must do a **./bootstrap** and **./configure −−enable−v4l −−with−ffmpeg=PATH** (where *PATH* is the absolute path to the directory where you compiled ffmpeg, and not the relative path !) .

There is no command line interface for this input for the moment, so you will have to use the configuration file .

Here is a typical configuration file `vls.cfg`to run VLS with a v4l device :

```
# vls.cfg (VLS configuration file)
# Example of the VideoLAN HOWTO for the Video for Linux input

BEGIN "Inputs"

  webcam       = "v4l"

END

BEGIN "webcam"

  # V4L device
  Device        = "/dev/video"

  # Channel Source : Webcam is often "0"
  #                  TV Card, may be Tuner(0), Composite(1), Svideo(2)
  Channel       = "0"

  # Norm: PAL=0, NTSC=1, SECAM=2
  Norm          = "0"

  # Frequency of the channel in Mhz, multiplied by 16 (tuner cards only)
  Frequency     = "7668"

  # Size possibilities: can be either:
  # − empty string : default size
  # − width x height
  # − subQCIF(128x96), QSIF(160x120), QCIF(176x144), SIF(320x240),
  #   CIF(352x288), VGA (640x480)
  Size          = ""

  # DeInterlace: "0" = no ; "1" = yes
  DeInterlace   = "0"

  # Video compression format: "mpeg1" or "mpeg4"
  Compression   = "mpeg4"

  # Video bitrate (approx.) in kbps
  Bitrate       = "500"

  # Quality: "1.0" (good) to "31.0" (bad)
  Quality       = "1.0"
```

```
  # Sound device
  AudioDevice   = "/dev/dsp"

  # Audio compression format: "mp2", "mp3" or "ogg"
  AudioCompression = "mp2"

  # Audio bitrate in kpbs
  AudioBitrate  = "64"

  # Audio frequency ("16000", "22050", "24000", "32000", "44100" or "48000")
  AudioFreq     = "16000"

  # Audio channel: "1" = mono ; "2" = stereo
  AudioChannel  = "2"

END

BEGIN "Channels"

  channel1       = "network"

END

BEGIN "channel1"

  # Unicast or multicast IP address
  DstHost = "192.168.0.42"

  # Destination port
  DstPort = "1234"

  # If it's a multicast IP address, uncomment the line below
  #Type    = "multicast"

  # If it's a multicast IP address, set the "Time To Live" below
  #TTL     = "12"

END

BEGIN "LaunchOnStartUp"

  command1 = "start video channel1 webcam"

END
```

Once you have adapted the configuration file above, run VLS :

```
% vls -vv -f vls.cfg
```

Chapter 9. Advanced use of VLC's stream output (transcoding, multiple streaming, etc...)

9.1. The syntax

Please refer to the *command line interface* chapter of the VLC user guide to learn the syntax of VLC's stream output. You can find the VLC user guide on the documentation page .

9.2. Examples

9.2.1. Transcoding

Transcode the input stream and send it to a multicast IP address with the associated SAP announce :

```
% vlc -vvv input_stream --sout '#transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128,deinterlace}:
standard{access=udp,mux=ts,url=239.255.12.42,sap=TestStream}'
```

Display the input stream, transcode it and send it to a multicast IP address with the associated SAP announce :

```
% vlc -vvv input_stream --sout '#duplicate{dst=display,dst=
"transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128,deinterlace}:
standard{access=udp,mux=ts,url=239.255.12.42,sap=TestStream}"}'
```

Transcode the input stream, display the transcoded stream and send it to a multicast IP address with the associated SAP announce :

```
% vlc -vvv input_stream --sout '#transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128,deinterlace}:
duplicate{dst=display,dst=standard{access=udp,mux=ts,url=239.255.12.42,sap=TestStream}}'
```

9.2.2. Multiple streaming

Send a stream to a multicast IP address and a unicast IP address :

```
% vlc -vvv input_stream --sout '#duplicate{dst=
standard{access=udp,mux=ts,url=239.255.12.42,sap=TestStream},
dst=standard{access=udp,mux=ts,url=192.168.1.2}}'
```

Display the stream and send it to two unicast IP addresses :

```
% vlc -vvv input_stream --sout '#duplicate{dst=display,dst=
standard{access=udp,mux=ts,url=192.168.1.12},
dst=standard{access=udp,mux=ts,url=192.168.1.42}}'
```

9.2.3. Transcoding and multiple streaming

Transcode the input stream, display the transcoded stream and send it to a multicast IP address with the associated SAP announce and an unicast IP address :

```
% vlc -vvv input_stream --sout '#transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128,deinterlace}:
duplicate{dst=display,dst=standard{access=udp,mux=ts,url=239.255.12.42,sap=TestStream},
dst=standard{access=udp,mux=ts,url=192.168.1.2}}'
```

Display the input stream, transcode it and send it to two unicast IP addresses :

```
% vlc -vvv input_stream --sout '#duplicate{dst=display,
dst="transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128}:
duplicate{dst=standard{access=udp,mux=ts,url=192.168.1.2},
dst=standard{access=udp,mux=ts,url=192.168.1.12}"}'
```

Send the input stream to a multicast IP address and the transcoded stream to another multicast IP address with the associated SAP announces :

```
% vlc -vvv input_stream --sout '#duplicate{dst=
standard{access=udp,mux=ts,url=239.255.1.2,sap=OriginalStream},
dst="transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128}:
standard{access=udp,mux=ts,url=239.255.1.3,sap=TranscodedStream}"}'
```

9.2.4. HTTP streaming

Stream in HTTP :

- on the server, run :
  ```
  % vlc -vvv input_stream --sout '#standard{access=http,mux=ogg,url=server.example.org:8080}'
  ```
- on the client(s), run :
  ```
  % vlc http://server.example.org:8080
  ```

Transcode and stream in HTTP :

```
% vlc -vvv input_stream --sout '#transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128}:
standard{access=http,mux=ogg,url=server.example.org:8080}'
```

For example, if you want to stream an audio CD in Ogg/Vorbis over HTTP :

```
% vlc -vvv cdda:/dev/cdrom --sout '#transcode{acodec=vorb,ab=128}:
standard{access=http,mux=ogg,url=server.example.org:8080}'
```

9.2.5. Use the *es* module

Separate audio and video in two PS files :

```
% vlc -vvv input_stream --sout
'#es{access=file,mux=ps,url_audio=audio-%c.%m,url_video=video-%c.%m}'
```

Extract the audio track of the input stream to a TS file :

```
% vlc -vvv input_stream --sout '#es{access_audio=file,mux_audio=ts,url_audio=audio-%c.%m}'
```

Stream in unicast the audio track on a port and the video track on another port :

- on the server side :

```
% vlc -vvv input_stream --sout
'#es{access=rtp,mux=ts,url_audio=192.168.1.2:1212,url_video=192.168.1.2:1213}'
```

- on the client side: to receive the audio :

  ```
  % vlc udp:@:1212
  ```

  to receive the video :

  ```
  % vlc udp:@:1213
  ```

Stream in multicast the video and dump the audio in a file :

```
% vlc -vvv input_stream --sout '#es{access_video=udp,mux_video=ts,url_video=239.255.12.42,
access_audio=file,mux_audio=ps,url_audio=audio-%c.%m}'
```

You can also combine the *es* module with the other modules to set–up even more complex solution .

Chapter 10. Stream in IPv6

10.1. Stream in IPv6

You will need an IPv6−aware operating system, like Linux 2.4.21 with the *ipv6* module loaded, Windows XP Service Pack 1 or Mac OS X version 10.2 or higher. Please look at the features pages on the VideoLAN web site to know about the status of IPv6 in VLC and VLS for each O.S .

If you are using VMware under Linux, you will have to stop VMware and unload the VMware kernel modules, because we noticed it prevented IPv6 streaming !

10.1.1. Stream with VLC

```
% vlc −vvv video1.xyz −−ipv6 −−sout udp:[ff08::1] −−ttl 12
```

where :

- `video1.xyz` is the file you want to stream (you can also put **dvdold:/dev/dvd** to stream a DVD or any other input configuration) ,
- *ff08::1* is either :
  - the IPv6 address of the machine you want to unicast to ;
  - or the multicast IPv6 address .
- *12* is the value of the TTL (Time To Live) of your IP packets (which means that the stream will be able to cross 11 routers) .

Under Unix/Linux, you may have to protect the square brackets around the IPv6 address :

```
% vlc −vvv video1.xyz −−ipv6 −−sout udp:\[ff08::1\] −−ttl 12
```

You may have to specify the output network interface :

```
% vlc −vvv video1.xyz −−ipv6 −−sout udp:[ff08::1%eth0] −−ttl 12
```

where *eth0* is the name of the network interface (under Linux the network interfaces are named *ethX*, under Mac OS X it's *enX* and under Windows it's *X*, where *X* is the appropriate number) .

10.1.2. Stream with VLS

You will need to use the configuration file `vls.cfg`. Please, see the VLS user guide to know how to write one for IPv6 streaming in unicast or multicast .

10.2. Receive in IPv6

10.2.1. Receive an unicast stream

```
% vlc −vvv −−ipv6 udp:
```

10.2.2. Receive a multicast stream

```
% vlc -vvv --ipv6 udp:@[ff08::1]
```

Under Unix/Linux, you may have to protect the square brackets around the IPv6 address :

```
% vlc -vvv --ipv6 udp:@\[ff08::1\]
```

You may have to specify the output network interface :

```
% vlc -vvv video1.xyz --ipv6 --sout udp:[ff08::1%eth0] --ttl 12
```

where *eth0* is the name of the network interface (under Linux the network interfaces are named *ethX*, under Mac OS X it's *enX* and under Windows it's *X*, where *X* is the appropriate number) .

Chapter 11. Video On Demand

11.1. Overview

With Video On Demand (VOD), the user can start the video when he wants, make pauses, go forward and back in the video. It is of course the best in video streaming and the dream for every user.

VOD is a very big consumer of resources for the server and the network. VOD is unicast, not multicast : this means that the network and server resources needed are directly proportional to the number of clients.

The design of VideoLAN's VOD solution is very simple. The idea is to do HTTP streaming, i.e. stream an MPEG video encapsulated in HTTP. The regulation of the bitrate between the client and the server is done automatically by TCP. With HTTP version 1.1, there is the possibility to seek in a file downloaded, that's what we use to seek in the video.

11.2. On the server side

On the VOD server, you need a running Web server. For example, you can use a Linux server running Apache. Other operating systems and other Web servers should work too, but we have never tested.

Make your MPEG−1, MPEG−2 or MPEG−4 / DivX files available to the clients on the Web server.

For example, we have a Web server whose DNS name is *localserver*. On this server, we put an MPEG file `video1.mpg` which will be available to the clients at the URL `http://localserver/video1.mpg`.

11.3. On the client side

```
% vlc −vvv http://localserver/video1.mpg
```

VLC starts to read the stream nearly immediately and you can seek in the stream, make pauses, etc... as if the stream was a local file.

Chapter 12. Add a channel information service

Typing multicast addresses is not very fun... that's why you need a channel information service ! VideoLAN has implemented a channel information service based on the SAP/SDP standard. The SAP announces about the multicast programs are sent on the network (via the multicast address *224.2.127.254* reserved for this purpose). On the client side, VLCs receive these annouces and automatically add the programs announced to their playlist.

VLC can send the SAP announces associated with the programs it is streaming. VLS can't do that, but an independant program, the mini−SAP−server, can send the announces for the programs sent by VLS.

12.1. Send announces when streaming with VLC

To send announces with VLC, you need to use the complex syntax of VLC's stream output, like this:

```
% vlc −vvv input_stream −−sout '#standard{access=udp,url=239.255.12.42,sap="Test Stream"}' −−ttl 12
```

where `video1.xyz` is the file you want to stream, *239.255.12.42* is the multicast IP address you want to stream on, *Test Stream* is the name that will be used for this program in the SAP announces and *12* is the value of the TTL (Time To Live) of the stream and of the SAP announces.

To do the SAP announces in IPv6, just add the *sap_ipv=6* option:

```
% vlc −vvv input_stream −−sout '#standard{access=udp,url=239.255.12.42,sap="Test Stream",sap_ipv=6}' −−ttl 12
```

12.2. Send announces for the programs streamed by VLS with the mini−SAP−server

The mini−SAP−server is only available for Linux and Mac OSX.

12.2.1. Install the mini−SAP−server

Download the latest version of the mini−SAP−server from the streaming download page.

Install−it:

```
% tar xvzf miniSAPserver−version.tar.gz
% cd miniSAPserver−version
% ./configure (if version >= 0.2.3)
% make
```

12.2.2. Configure the mini−SAP−server

Edit the configuration file `sap.cfg` shipped with the tarball. It should contain a global section with the Time To Live (TTL) and the IP version (IPv4 or IPv6) used for the SAP announces and a section per program announced. Use the comments to understand each parameter.

The Time To Live option indicates the maximum number of routers that the SAP announce packets can cross before being dropped.

12.2.3. Run the mini–SAP–server

Start the mini–SAP–server:

```
% ./sapserver -c sap.cfg
```

If you want to run the mini–SAP–server in the background, use the **–d** command line switch. For instance:

```
% ./sapserver -d -c sap.cfg
```

You can also omit the **–c** command line switch if your config file is `/etc/sap.cfg`.

12.3. Start VLC(s) on the client(s)

- If the announces are sent in IPv4 :
  ```
  % vlc -vvv --extraintf sap
  ```
- If the announces are sent in IPv6 :
  ```
  % vlc -vvv --extraintf sap --sap-ipv6
  ```

Then open the playlist: you should see the names of the programs announced in SAP. When you double–click on the name of a program, VLC will subscribe to the multicast address and start to play the stream !

Appendix A. GNU Free Documentation License

Version 1.2, November 2002

A.1. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

A.2. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world−wide, royalty−free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front−matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front−Cover Texts or Back−Cover Texts, in the notice that says that the Document is released under this License. A Front−Cover Text may be at most 5 words, and a Back−Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine−readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard−conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine−generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

A.3. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

A.4. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front−Cover Texts on the front cover, and Back−Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine−readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer−network location from which the general network−using public has access to download using public−standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

A.5. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
D. Preserve all the copyright notices of the Document.
E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.
I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front−matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties−−for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front−Cover Text, and a passage of up to 25 words as a Back−Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front−Cover Text and one of Back−Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

A.6. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

A.7. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

A.8. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

A.9. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

A.10. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

A.11. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

A.12. ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front−Cover Texts, and no Back−Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front−Cover Texts and Back−Cover Texts, replace the "with...Texts." line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front−Cover Texts being LIST, and with the Back−Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.