VLC Build

If you've already installed vlc remove it using the following:

```
Sudo apt purge vlc
Sudo apt autoremove
```

You have to create a directory in your home and within it a "build" directory for the build.
Then clone this: `git clone https://github.com/videolan/vlc.git`

This will create a directory called vlc with the vlc source code.

Then go to the vlc directory and change the version.
You should use version 3.0.18. Switch to the 3.0.x branch and checkout the correct tag:

1. `git switch 3.0.x`
2. `git checkout tags/3.0.18`

You need to edit the file `/etc/apt/sources.list` and, for each repository entry, enable the corresponding source repository by removing the "#" sign at the beginning of the entry. Each pair of binary and source repositories will resemble this:

```
deb http://gb.archive.ubuntu.com/ubuntu/ jammy main restricted
deb-src http://gb.archive.ubuntu.com/ubuntu/ jammy main
restricted
```

After the sources.list file is edited give this command: `sudo apt update`

This should be done for every repository pair which is enabled in this file.

Then run this command: `sudo apt build-dep vlc`

The Live555 package, needed for RTP/RTSP, requires more involvement:

1. Create a directory somewhere in your home directory, such as "live555" and enter it (you can call it anything you like, but throughout these instructions, "live555" will be the name assumed):
    a. `mkdir live555`
    b. `cd live555`
2. Get the source code from the official site using:
    a. `wget https://download.videolan.org/contrib/live555/live.2021.11.14.tar.gz`
        - You should use this version of Live555 with VLC, as its API is frequently changed. It seems other versions may work as well, but it is difficult to be sure without testing them.

    b. Extract the source code: `tar -xvf live.2021.11.14.tar.gz`. This will extract the source into a directory called “live”.
3. Enter the extracted directory: `cd live/`.
4. Build and install the package:
    a. `./genMakefiles linux`
        - This will generate makefiles for building a static library, rather than a shared library.
    b. `CFLAGS="-fPIC" CXXFLAGS="-fPIC" make -j 4`
        - The CFLAGS and CXXFLAGS parameters enable PIC (position independent code), which is necessary for building a static library that VLC can be compiled with.
        - It is possible to directly edit the makefile using an editor or the Sed program to enable PIC, but this is the most straightforward way.

It is not necessary to install Live555 with “make install” into a global system location, as you should instead explicitly specify the location of the libraries when configuring the build of VLC. This will make it easier to manage multiple builds of Live555, if it turns out to be necessary.

Enter the VLC source directory: `cd vlc/`.

Bootstrap the build system – run this command in the VLC source directory: `./bootstrap`

Run the configure script – this needs to be run with the required flags for enabling Live555: Change the path to the directory made for the build before.

```
./configure --prefix=/path/to/vlc/build/
LIVE555_CFLAGS="-I$HOME/live555/live/BasicUsageEnvironment/include
-I$HOME/live555/live/groupsock/include
-I$HOME/live555/live/liveMedia/include
-I$HOME/live555/live/UsageEnvironment/include"
LDFLAGS="-L$HOME/live555/live/BasicUsageEnvironment
-L$HOME/live555/live/groupsock -L$HOME/live555/live/liveMedia
-L$HOME/live555/live/UsageEnvironment"
LIVE555_LIBS="-Wl,--start-group -lliveMedia -lBasicUsageEnvironment
-lgroupsock -lUsageEnvironment -lssl -lcrypto -Wl,--end-group"
```

    - This assumes that you have built Live555 in a directory in your home directory called “live555”, within a subdirectory called “live”. You should change these flags if you have built Live555 in a different location.
    - Any circular dependencies are dealt with by the `-Wl,--start-group` and `-Wl,--end-group` options.
2. Compile VLC: `make -j 4`
3. Install VLC into the build directory: `make install`

You will find both `vlc` and `cvlc` in the `bin` directory:

```
peter@ubuntuvm:~/vlc/build/bin$ ls
cvlc  nvlc  qvlc  rvlc  svlc  vlc  vlc-wrapper
```

You can give this command to see if you have built the correct version: ./vlc --version

```
VLC media player 3.0.18 Vetinari (revision 3.0.18-0-ge9eceaed4d)
VLC version 3.0.18 Vetinari (3.0.18-0-ge9eceaed4d)
Compiled by lahirudissanayake on LAHIRU-D-L.zone24x7.lk (Jul  4 2023 14:26:54)
Compiler: gcc version 11.3.0 (Ubuntu 11.3.0-1ubuntu1~22.04.1)
This program comes with NO WARRANTY, to the extent permitted by law.
You may redistribute it under the terms of the GNU General Public License;
see the file named COPYING for details.
Written by the VideoLAN team; see the AUTHORS file.
```

The version should be 3.0.18

m

Streaming process

You hava to create a content directory in your home directory and run this command:

wget http://s3.amazonaws.com/akamai.netstorage/HD_downloads/Orion_SM.mp4
This will download the video for the streaming.

You can run this command to start the video streaming process:

```
./cvlc -vvv ~/content/Orion_SM.mp4 --sout '#rtp{port=554,mux=ts,sdp=rtsp://localhost:8080/stream.sdp}' --repeat
```

Run this command to start the video stream using http:

```
cvlc -vvv ~/content/Orion_SM.mp4 --sout '#standard{access=http,mux=ts,dst=:8080}' --repeat
```

Documentation for cvlc streaming

While this is streaming you can open another tab and run this commond in the vlc/build/bin directory:

./vlc -vvv rtsp://localhost:8080/stream.sdp

Add this line to the ~/.bashrc file to connect the vlc lib to be able to import:

Add this line before generating the python-vlc bindings:

```
# adding the vlc library path to the environment variable
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/home/nqosm/vlc/build/lib

#adding the python-vlc module path to the environment variable
export
PYTHONPATH=$PYTHONPATH:/home/nqosm/vlc/vlc/bindings/python/generated/
3.0
```