# XFCE's ALSA Plugin on Arch

How to build the default plugin on aarch64/ Raspberry Pi 4 64bit

Onkobu Tanaake

2020-10-23T10:00:00

XFCE has an ALSA plugin to control audio volume. It can be configured to use a certain device and works flawlessly. But it is not available through Arch's AUR (Arch User Repository) for aarch64. Raspberry Pi 3B and especially 4 support 64bit and therefore require compatible packages. The following sections guide you through the 15min process of compiling the plugin on the target system.

- `pacman -S base-devel graphviz meson ninja`
- wget and auto-build vala
- git clone xfce4-alsa-plugin and make/ make install
- provide symbolic links of plugin artifacts from

  `/usr/local`

  to Arch's XFCE-locations

First of all install the development packages on the target system for Arch Linux by issueing a . The second package is necessary as dependency for the upcoming Vala environment. The first package is a meta package. You need to specify which sub packages need to be installed. It is absolutely ok to choose all. Finally meson and ninja may be already part of base-devel. I put them explicitely there since base-devel could change in the future and come without ninja or meson.

Now you are able to compile two more packages on your Raspberry Pi 4 or any other aarch64. Start by downloading sources of Vala [https://wiki.gnome.org/Projects/Vala]. This is the development package for Gnome's object oriented C also used by XFCE developers. You can use the latest version. I used 0.50.1. Unpack the tar.xz-file, e.g. . Change into the directory that appeared. Follow the installation instructions described there in one of the files (README, INSTALL...). Basically some autotools commands like followed by to compile and as super user/ root to install the binaries and libraries.

The last package is the plugin itself. Get it from Github by cloning the repository [https://github.com/equeim/xfce4-alsa-plugin]. There is a green button *Code* that reveals the repository's URL. Use this with the command . The git-command is provided through base-devel meta package. Change into the cloned repository's directory. Follow the installation instructions of the Git repository. It uses meson/ ninja and is basically something like . I combined three commands with && to only execute the entire chain if all succeed. Mind that tries to write with super user permissions. Hopefully your user is allowed to sudo. If not, enable this temporarily or run only the install-command as root itself.

If all went fine and the last build and install commands of xfce4-alsa-plugin finished you have two important artifacts in -directory. This is absolutely correct but does not work yet. You will not be able to add the plugin to the XFCE-panel. Create symbolic links for the two files to the Arch structure as mentioned in XFCE's plugin development [https://wiki.xfce.org/dev/howto/panel_plugins]:

- `sudo ln -s /usr/local/lib/xfce4/panel/plugins/libalsa.so /usr/lib`
- `sudo ln -s /usr/local/share/xfce4/panel/plugins/alsa.desktop /usr`

The .desktop file registers the shared object/ library. Now you can add the new component to your panel. I also tried other options that either didn't compile at all or weren't provided for aarch64 at all. Try not to adjust the plugin's target directories when installing. This often breaks de-installing of Arch packages not being able to remove orphaned files/ directories. Instead go with the sort of overlay in /usr/local and symbolic links. It'd even be better to put all your locally compiled packages into /opt instead – an approach I prefer on all machines I like to have compliant to Linux Standard Base (LSB) – I'm one of the oldies remembering this. Look it up yourself how to configure the install prefix for the different build tools to use /opt instead of /usr(/local).