PulseAudio Rationale

With the release of the first beta of what will be version 14.2, PulseAudio has been introduced as default audio server in Slackware Linux.

This historical change comes out mostly from necessity, as the bluetooth stack 🎧BlueZ has dropped the ALSA plugin for audio output, making the presence of PulseAudio mandatory for playing any audio coming from Bluetooth devices.

From Slackware's Changelog:

```
After upgrading to BlueZ 5 recently, everything seemed to be working great,
but then it was pointed out that Bluetooth audio was no longer working.
The reason was that the newer BlueZ branch had dropped ALSA support and now
required PulseAudio. So with some trepidation, we began investigating adding
PulseAudio to Slackware. Going back to BlueZ 4 wasn't an option with various
dependent projects either having dropped support for it, or considering
doing
so. After several iterations here refining the foundation packages and
recompiling and tweaking other packages to use PulseAudio, it's working well
and you'll likely not notice much of a change. But if you're using Bluetooth
audio, or needing to direct audio through HDMI, you'll probably find it a
lot
easier to accomplish that.
Best of all, we're finally a modern, relevant Linux distro! ;-)
```

Before this moment, there was no technical need for introducing PulseAudio as a basic component of the distribution, and linking the audio components directly to ALSA devices gave all the features needed by users without PA's known backsides.

Removing PulseAudio Completely

It should be noted that, as PulseAudio is now a basic system component, with many applications directly linked to it, the following procedure for removing PulseAudio is unsupported. It did work for me, though.

In order to remove PulseAudio from your system completely, follow these steps:

```
su
mv /etc/asound.conf /etc/asound.conf.disabled
removepkg pulseaudio
```

However, after you do this, you will find many applications (mplayer etc) no longer work, because they are linked to libpulse.so.0. This issue is remedied with the following:

```
git clone https://github.com/i-rinat/apulse
cd apulse
```

```
mkdir build
cd build
cmake -DCMAKE_INSTALL_PREFIX=/usr -DCMAKE_BUILD_TYPE=Release ..
make
su
mv libpulse* /usr/lib64 #or /usr/lib if you are on a 32-bit machine
```

Now your pulseaudio-linked programs will find a fake libpulse.so.0 that nevertheless provides all the symbols they expect and everything that can use ALSA will use ALSA just like it used to. Additionally, any program insisting on trying to use the libpulse.so.0 API should theoretically still work as the libpulse.so.0 you have installed is a wrapper to make PulseAudio programs (Skype specifically) work on ALSA-only systems.

Using Pulseaudio as non-default audio manager

PulseAudio has become the default audio manager in the distribution, and it is configured for taking ownership of the audio card in the system so that all the components needing access to audio must refer to it.

For this reason, multiple software packages have already been recompiled to enable dependency on PulseAudio, and others will probably follow before the release cycle of 14.2 is complete. This equates Slackware to all other Linux distributions that had previously adopted PulseAudio.

The goal of this guide is giving instructions on how to configure PulseAudio in Slackware-current and upcoming 14.2 as an additional audio system while keeping direct access to ALSA devices as the default.

It is possible to configure PulseAudio as a "dumb pipe" that streams audio to ALSA's dmix device, without directly grabbing the audio card and allowing other software to stream into dmix device as well, so that all audio sources are mixed together.

I will also try to keep track of the changes in compilation parameters of various packages that must be reverted in order to have them use ALSA as their default output.

The final result should be having /usr/bin/pulseaudio process started as less as possible, ideally only when the components that cannot work without it are in use, while the rest of the system will use ALSA.

Basic Pulseaudio configuration

Steps to revert ALSA configuration to playing directly to audio hardware instead of using Pulseaudio are:

- Remove /etc/asound.conf file that has been installed by the **alsa-lib** package

> You may want to create a dummy /etc/asound.conf file so that future upgrades of alsa-lib package will not install it again and will rather place a /etc/asound.conf.new

- Remove **alsa-plugins** package

```
# removepkg alsa-plugins
```

The rationale for this is:

```
l/alsa-plugins-1.1.0-x86_64-1.txz: Added.
        This provides a plugin to redirect ALSA output to PulseAudio.
```

- Edit /etc/pulse/default.pa
  Find and uncomment lines which load back-end drivers. Add device parameters as follows. Then find and comment lines which load autodetect modules.

```
load-module module-alsa-sink device=dmix
load-module module-alsa-source device=dsnoop
# load-module module-udev-detect
# load-module module-detect
```

- Edit /etc/pulse/daemon.conf
  - Add configuration for having Pulseaudio daemon immediately exit as soon as the audio source triggering it has ended:

```
exit-idle-time = 0
```

  - Optional: add configuration for keeping Pulseaudio from altering volume levels that ALSA is handling

```
flat-volumes = yes
```

Changes related to X11

The **pulseaudio** package will install /etc/xdg/autostart/pulseaudio.desktop. This file contains configuration responsible for registering Pulseaudio with the current X11 session and making it available to all programs running in X11 environment.

An XSMP connection between Pulseaudio daemon and the X11 session will be created, and this will make the daemon stay started during all the session, even if no Pulseaudio clients are actually in use.

```
$ pactl list clients

Client #0
        Driver: module-x11-xsmp.c
```

```
        Owner Module: 4
        Properties:
                application.name = "XSMP Session on KDE as
10d7cad6d00001453223274000000022300109"
                xsmp.vendor = "KDE"
                xsmp.client.id = "10d7cad6d00001453223274000000022300109"
```

But without this connection no audio software running in the graphical session will be able to connect to Pulseaudio daemon even if it correctly starts on demand.

If you are fine with having `/usr/bin/pulseaudio` running idle in the system and being occasionally used by the clients depending on it, you can leave things untouched.

If you want it to start and stay running only for the time it's actually needed and exit right after closing any related client then you can take this solution into consideration:

remove or rename the file `/etc/xdg/autostart/pulseaudio.desktop`. This will avoid PA being pre-started at each user login to an X11 session.

Create a script `/usr/local/bin/pa-unreg` with the following:

```bash
#!/bin/bash
if [ "$(pidof pulseaudio)" ]
then
        /usr/bin/pactl unload-module module-x11-xsmp 1>/dev/null 2>/dev/null
fi
```

Install the following line in user's personal crontab:

```
$ crontab -l
# Unregister Pulseaudio XSMP module
*/10 * * * * /usr/local/bin/pa-unreg
```

This command running every 10 minutes will unregister any Pulseaudio daemon currently running from the XSMP session. When Pulseaudio daemon starts it will initially register itself and the client triggering it will be able to connect. Then within the following 10 minutes the registration will be removed, but currently established client connections will stay active until when necessary. When clients will exit, the daemon will exit as well because no kind of session is still alive.

> You may be using applications that can dynamically detect the presence of Pulseaudio daemon and connect to it, while they fallback to using ALSA when it's not detected. For example, multimedia components of Mozilla Firefox and Google Chrome have this behaviour. If you are using any of these programs during the time that PA's XSMP registration is active, they will open a session with it and they will trigger it to stay started even after the session is unregistered and other audio clients have been closed.

Changes related to Desktop Environments

KDE 4

As of the current date (January 2016), Slackware-current and upcoming release 14.2 include KDE 4.14.3. This version is known as being the last official release of KDE 4. No further releases are planned, and no further packages should arrive in the distribution, unless any recompilation should be required because of dependency changes.

With the arrival of Pulseaudio, some packages related to KDE have been recompiled to enable PA support by linking them to its libraries.

Notably these packages are:

- **phonon-4.8.3** (the abstraction layer for multimedia streams)
- **kmix-4.14.3** (the mixer for handling volume levels)
- **kde-runtime-4.14.3** (probably for the Control Panel tabs related to audio configuration)

This means that as soon as the user logs into a KDE session the `/usr/bin/pulseaudio` will be started and will stay running even after we have applied the previously listed changes.

Phonon

It handles all audio streams coming from KDE programs (including Amarok, for example). Now it is linked to Pulseaudio libraries and sends all them through it. Reverting it to the previous condition will make all KDE multimedia applications go through ALSA again, without any individual change.

There is no runtime option for making it use ALSA output when it's compiled for Pulseaudio, so you will need to recompile it with explicit exclusion of Pulseaudio support, and then replace the official Slackware package with your own.

You will need this addition to CMake configuration parameters:

```
cmake \
    [...]
    -DWITH_PulseAudio=FALSE \
    -DPHONON_PULSESUPPORT=OFF \
    [...]
```

You can replace the PA-dependant package with this other one with the command:

```
# upgradepkg --reinstall phonon-4.8.3-x86_64-1.txz
```

KMix

The audio mixer has been recompiled for controlling volume levels provided by Pulseaudio devices,

but in this case it is possible to configure it at runtime for hooking to ALSA devices like it did in the past.

The required configuration is:

```
$ echo export KMIX_PULSEAUDIO_DISABLE=1 > ~/.kde/env/kmix_disable_pulse.sh
$ chmod +x ~/.kde/env/kmix_disable_pulse.sh
```

The executable script `kmix_disable_pulse.sh` will be started at user's login to alter the default behaviour.

KDE 5

KDE 5.x is currently not officially part of Slackware Linux. It is unofficially provided and mantained by Eric Hameleers, and now it has been configured for using Pulseaudio by default.

For the goals of this guide, the required steps for having it use ALSA as standard output device are the same as for KDE 4.

Please notice that environment startup scripts that in KDE4 were previously placed in `~/.kde/env` must now be placed in `~/.config/plasma-workspace/env`, so in order to configure KMix for ALSA mixer management you will now have:

```
$ echo export KMIX_PULSEAUDIO_DISABLE=1 > ~/.config/plasma-
workspace/env/kmix_disable_pulse.sh
$ chmod +x ~/.config/plasma-workspace/env/kmix_disable_pulse.sh
```

> ⚠️ Remember to avoid using the PulseAudio based volume control Plasmoid for your system tray.

Configuration of specific applications

Some of the multimedia software present in Slackware can be configured for using ALSA output or PulseAudio output at run-time, but there is software where the output selection is hardcoded and it can be changed only by means of a recompilation.

All the software that is using KDE's phonon for their output (eg: Amarok) will not require any further changes besides those listed above.

Here is an evolving list of common software and required actions:

mpg123

This software has been compiled for using Pulseaudio as default output but you can change this by simply adding an option on command-line:

```
$ mpg123 <...> -o alsa <...> file.mp3
```

Clementine

Clementine audio player requires being compiled with the following cmake options:

```
cmake \
    [...]
    -DENABLE_LIBPULSE=NO
    [...]
```

Audacious

Also Audacious can be configured to use ALSA even if it's been compiled with PA support.

In Settings→Audio you can define Output Plugin to ALSA Output.

Sources

* Originally written by LoneStar

howtos, pulseaudio, sound

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/es:howtos:multimedia:pulseaudio_non-default

Last update: **2019/02/28 02:04 (UTC)**