Better Music With Free Software

BV

[2014-12-31 Wed 13:57]

1 Overview

The house has a couple of audio receivers that can take the usual gamut of analogue as well as optical S/PDIF and one can also take HDMI inputs. There are also a collection of powered computer speakers.

On the computer front, there is a relatively powerful desktop workstation (`haiku`, i7) and an older desktop (`hype`, Athlon II), a few laptops, a Raspberry Pi (`rpi`, rev B) and a BeagleBone Black (`bbb`, rev C). All computers run some flavor or derivative of Debian.

Little gizmos include a half dozen Android devices in the form of phones, ex-phones and tablets and two Roku.

The goal here is to have a way to play audio throughout the house with these features:

- Central location for audio files.
- Synchronized play on multiple "receivers".
- Limiting to one receiver.
- Control via Android.

2 Streaming methods

2.1 MPD + RTP + Pulse

Try following this work:

- `http://www.hackerposse.com/~rozzin/journal/whole-home-pulseaudio.html`

- https://fruit.je/mpd-rtp
- http://www.freedesktop.org/wiki/Software/PulseAudio/Documentation/User/Network/RTP/
- http://nickschicht.wordpress.com/2013/08/15/raspberry-pi-mpd-streaming-using-puls
- http://anarcat.koumbit.org/2013-02-03-live-radio-streaming-mpd-part-1-multicast-r
- thread with main players linked

2.1.1 Overview

- Run MPD on one server
- Run Pulse on receivers

2.1.2 Server

On haiku:

```
# apt-get install mpd
```

In file:///etc/mpd.conf add/change:

```
bind_to_address         "0.0.0.0"

audio_output {
            type            "pulse"
            name            "MPD Stream"
            sink            "rtp"
            description     "RTP from MPD"
            mixer_type      "software"
}
```

In file:///etc/pulse/default.pa add

```
load-module module-null-sink sink_name=rtp format=s16be channels=2 rate=44100
load-module module-rtp-send source=rtp.monitor
```

Also symlink some audio directories into /var/lib/music.

2.1.3 Receiver

On `rpi` (running `2014-12-24-wheezy-raspbian.img`).

```
# apt-get update
# apt-get install pulseaudio pavucontrol
```

And, add to its `/etc/pulse/default.pa`

```
load-module module-rtp-recv
```

2.1.4 Conclusions

I can get the server side working so that I can use mplayer to play on `rpi`. The play back is sometimes choppy. This is fully on GbE wire. From a wireless laptop there is more dropout than audio. This sucks.

I can get `rpi` pulseaudio working to play with `pacmd play-file <file> <sink>`. I can also see pulseaudio accepting RTP traffic but I can not figure out how to make it forward this to the actual sound card sink. Double sucks.

Try another tack.

2.2 p4sync

`https://snarfed.org/synchronizing_mp3_playback`

Build issues

2.3 SyncPlay

`http://syncplay.pl/`

Meant for videos. Meant to sync well enough for people to irc/mumble about a shared viewing of a video. Unclear what level of sync that is but I suspect not good enough.

2.4 Logitech Media Streamer

Free of charge server, Free Software clients (Squeezelite)

`http://www.mysqueezebox.com/download`

2.5 Gstreamer

See `./gstreamer.org`.

2.6 Icecast

2.7 Jack

2.8 Liquidsoap

3 Timing

The method requires each computer playing a stream to have its clock synchronized with the server. Their absolute time accuracy is not a concern. Keeping computer clocks synced to a second is easy. Getting that sync down to a millisecond become a challenge. There is lots of work on this topic:

- NTP FAQ entry defining terms of the art and with some analysis of clocks

3.1 Chrony

I compared the clocks on the RPI and the BBB using `clockdiff` run from the media server. With both running NTP, the RPi seems synced and stable but the BBB is way off and drifts up and down. I switched all three computers to Chrony and ran `clockdiff` against the two overnight. The results are shown below. The BBB is black and the RPi is red (as is fitting!).

Full series [width=.9]./sync

Zoom into the start. I don't know what that big jump is but `ntpdate` is still installed on the media server so it may have had it's own clock updated at that point. [width=.9]./sync-zoom-to-start

Zoom the Y-axis to better see long-term small effects. Eventually the sync within 1ms or less. [width=.9]./sync-zoom-long-term

After a couple of hours, the long-term timing measurements from these plots are given in the table below:

measure	rpi (ms)	bbb (ms)
full avg	0.10	0.98
full RMS	0.97	4.42
l.t. avg	0.03	-0.19
l.t. RMS	0.42	0.42

Here, "full" means the entire time while "l.t." means the long term measure after about two hours.

4 PTPd

Precision Time Protocol is for sync across the LAN.

The version on the rpi/bbb is 2.3.0 while on Ubuntu 14.04 it is 2.2.2. The command line options have changed between versions.

Following this blog entry I test on rpi/bbb (2.3.0) with:

```
$ sudo ptpd -b eth0 -g -C -D -E --e2e
```

and on the server (2.2.2) with:

```
$ sudo ptpd -G -b eth0 -c -D
```

That seems to go well and checking with `clockdiff` gives mostly 0's.

Make it stick in `/etc/default/ptpd` and start the daemons proper. On rpi/bbb:

```
START_DAEMON=yes
PTPD_OPTS="-i eth0 -s -E --e2e -f /var/log/ptpd.log"
```

Note, in trying to add `"-D"` on bbb it shows:

```
Runtime debug not enabled. Please compile with RUNTIME_DEBUG
```

Trying `"-V"` is too verbose for logging but it does who high precision offsets.

On server:

```
START_DAEMON=yes
PTPD_OPTS="-b eth0 -G"
```

This brings rpi/bbb solidly into the 0ms bin according to `clockdiff`.

5 Debian on BBB

5.1 Initial install

- Got wheezy image

from `http://beagleboard.org/latest-images`

- `xz -cb image.xz > /dev/mmcblk0`
- Move to BBB, power up with user/boot button pressed
- can log in via SSH almost immediately (`root` and no

pass-
word)
but
wait
for
flash
to
fin-
ish.
The
BBB
will
power
down

- remove SD card and power back up

5.2 Pair down

Remove all stuff related to running an X11 server session.

```
# apt-get remove --purge xserver-xorg-core
# apt-get remove --purge  lxpanel lxsession lxterminal lxmenu-data lxde-core lxde-comm
# apt-get remove --purge gnome-icon-theme gnome-keyring gnome-themes-standard gnome-th
# apt-get remove --purge gsettings-desktop-schemas xscreensaver xscreensaver-data xser
# apt-get autoremove --purge
```

5.3 Upgrade to jessie

- edit `/etc/apt/sources.list` and add

a jessie line for each existing wheezy one. Note, the beaglboard.org one doesn't have a jessie counterpart.

- apt-get dist-upgrade
- apt-get autoremove --purge
- apt-get clean
- reboot

Note: some problem upgrading udev.
Remove all GStreamer 0.10 and install

```
# apt-get install libgstreamer1.0-dev gstreamer1.0-tools gstreamer1.0-plugins-base libg
```

```
# apt-get install autoconf automake
# apt-get install libjson-glib-dev libavahi-glib-dev libsoup2.4-dev libgtk-3-dev
```

6 Build aurena

```
$ git clone https://github.com/thaytan/aurena.git
$ cd aurena
$ ./autogen.sh
$ make
```