A Headless Mumble Client for Single Board Computers, PC's or Virtual Environments

An IP Intercom / Transceiver / Gateway

talKKonnect (www.talkkonnect.com) is a headless self contained Mumble Push to Talk (PTT) client with a mobile transceiver form factor, complete with LCD, channel, volume and many other controls. This project is a fork of talkiepi by Daniel Chote. You can find Daniel's page here http://projectable.me . talKKonnect was developed initially to run on Raspberry Pi 3b+, but it can be successfully built on other boards, like Orange Pi. talKKonnect can also run in virtual environments (Oracle VirtualBox, KVM and Proxmox).



Hardware Improvements

You can use external microphone and push buttons on the device for Channel Up/Down navigation for radio-like experience. talKKonnect works with 4×20 Hitachi HD44780 LCD screen. Other screens like OLED will also be supported at a later stage. Low cost audio amplifiers like PAM8403 or similar "D" class amplifiers, can be used with talKKonnect builds. talKKonnect functions can also be controlled by key shortcuts from a terminal console screen.




4 LED's show online status, if any other participants have joined the talk group, when talKKonnect unit transmits and when the other clients are talking. talKKonnect can be used as a communications bridge to the external systems, like radio networks. It can be easily interfaced to portable or base radios (Beefing portable radios or UART radio boards). talKKonnect can be used with low cost GPS dongles (for instance "u-blox") for GPS tracking. talKKonnect has a "panic button" feature. When the panic button is pressed, talKKonnect will send an alert message with

GPS grid to the talk group, followed by an email. talKKonnect can also automatically send an audio stream on such an event.

talKKonnect radio gateway

Software Improvements

Colorized LOG is shown on the debugging terminal for events as they happen. talKKonnect will play alert sounds as different events happen. talKKonnect supports TTS prompts and can announce different events. It supports "Roger Beeps" playing on release of the PTT button. The speaker is muted when pressing PTT to prevent feedback and give a radio like experience. LCD is showing useful channel information, server information, channels, who joined, who is speaking, etc. talKKonnect configuration is kept in a single highly "granular" XML file, where many of its different options can be enabled or disabled.

Installation Instructions

1. Download the latest version of Raspbian Stretch Lite from https://www.raspberrypi.org/downloads/raspbian . At the time of making this document latest image release date was 2018-11-13 (Kernel Version 4.14). Download the ZIP file and extract IMG file to some temporary directory.

2. Use any USB / SD card imaging software for Windows of your other OS. Some of the many options are:

USB Image Tool:	https://www.alexpage.de/usb-image-tool
Win32 Disk Imager:	https://sourceforge.net/projects/win32diskimager
Rufus:	https://rufus.ie
Etcher:	http://www.etcher.io
Linux dd tool:	https://elinux.org/RPi_Easy_SD_Card_Setup

3. After the imaging, insert the SD card into your Raspberry Pi 3 b+, connect the screen, keyboard and power supply and boot into the OS.

4. Log in as user "pi" with password "raspberry" (this is the default username and password for a fresh install of Raspbian)

5. Do a `sudo passwd root` to set the new root password. Log out of the account pi and log into the root account with your newly set password

6. run `raspi-config` and expand the file system by chosing *"Advanced Options"->"Expand File System"*. Reboot.

7. Next go to *"Interfacing Options"* in raspi-config and *"Enable SSH Server"*.
Edit the file `/etc/ssh/sshd_config` with nano editor. Change the line
`#PermitRootLogin prohibit-password` to
`PermitRootLogin Yes`
Restart ssh server with
`service ssh restart`
Now you should be able to log in remotely via ssh using the root account and continue the installation.

8. Add user "talkkonnect"
```
adduser --disabled-password --disabled-login --gecos "" talkkonnect
```

9. Add user "talkkonnect" to groups
```
usermod -a -G cdrom,audio,video,plugdev,users,dialout,dip,input,gpio
talkkonnect
```

10. Update Raspbian
```
apt-get update
```

11. Install prerequisite programs
```
apt-get install golang libopenal-dev libopus-dev libasound2-dev git
ffmpeg omxplayer screen
```
(Note: If building talkkonnect on other than Raspberry Pi board, install mplayer instead of omxplayer)

12. Decide if you want to run talkkonnect as a local user or root? Up to you.
To build as a local user
```
su talkkonnect
```
(Note: you can also build talKKonnect as root, if you prefer).

13. Create code and bin directories
```
cd /home/talkkonnect
mkdir /home/talkkonnect/gocode
mkdir /home/talkkonnect/bin
```

14. Export GO paths
```
export GOPATH=/home/talkkonnect/gocode
export GOBIN=/home/talkkonnect/bin
```

15. Get programs and prepare for building talKKonnect
```
cd $GOPATH
go get github.com/talkkonnect/talkkonnect
cd $GOPATH/src/github.com/talkkonnect/talkkonnect
```

16. Before building the binary, confirm all features which you want enabled, the GPIO pins used and talKKonnect program configuration by editing file: `/home/talkkonnect/gocode/src/github.com/talkkonnect/talkkonnect/talkkonnect.xml` .

17. Build talKKonnect and test connection to your Mumble server.

```
go build -o /home/talkkonnect/bin/talkkonnect cmd/talkkonnect/main.go
```

18. Start talkkonnect binary

```
cd /home/talkkonnect/bin/talkkonnect
./talkonnect
```

19. Or create a start script

```
cd
sudo nano talkkonnect-run
```

with content:

```
#!/bin/bash
killall -vs 9 talkkonnect
sleep 1
reset
sleep 2
/home/talkkonnect/bin/talkkonnect
```

Make the script executable

```
chmod +x talkkonnect-run
```

19. You can start talKKonnect automatically on Raspberry Pi start up with “screen” program help. Add this line to /etc/rc.local file. before “exit 0”:

```
screen -dmS talkkonnect-radio /root/talkkonnect-run &
```

Then connect to active screen session with command “`screen -r`”. Exit the screen session with “Ctrl-A-D”.

```
 _        _ _    _                                     _
| |_ __ _| | | _| | _____  _ __  _ __   ___  ___| |_
| __/ _` | | |/ / |/ / _ \| '_ \| '_ \ / _ \/ __| __|
| || (_| | |   <|   < (_) | | | | | | |  __/ (__| |_
 \__\__,_|_|_|\_\_|\_\___/|_| |_|_| |_|\___|\___|\__|

A Flexible Headless Mumble Transceiver/Gateway for RPi/PC/VM

Created By : Suvir Kumar  <suvir@talkkonnect.com>

Version 1.32 Released January 2 2019
Additional Modifications Released under MPL 2.0 License

visit us at www.talkkonnect.com and github.com/talkkonnect
Press the <Del> key for Menu Options or <Ctrl-c> to Quit talkkonnect
```

talKKonnect welcome screen

Audio configuration

USB Sound Cards

For your audio input and output to work with talKKonnect, you needs to configure your sound settings. Configire and test your Linux sound system before building talKKonnect. talKKonnect works well with ALSA. There is no need to run it with PulseAudio. Any USB Sound cards supported in Linux, can be used with talKKonnect. Raspberry Pi's have audio output with BCM2835 chip, but unfortunately no audio input, by the design. This is why we need a USB sound card. Many other types of single board computers come with both audio output and input (Orange Pi). USB Sound cards with CM sound chips like CM108, CM109, CM119, CM6206 chips are affordable and very common.

When connected to a Raspberry Pi, USB sound card can be identified with "`lsusb`" command. Typical response is something like this:

```
Bus 001 Device 004: ID 0d8c:000c C-Media Electronics, Inc. Audio Adapter
```

Audio playback devices can be listed with "`aplay -l`" command.

Optional: When external USB Sound card is used, Raspberry Pi BCM2835 internal sound can be blacklisted or preveneted to load. To disable BCM2835 sound:

```
nano /boot/config.txt
```

Add this line:

```
#Disable audio (loads snd_bcm2835)
dtparam=audio=off
```

Save file and reboot.

If the BCM2835 sound is kept enabled, the USB sound card will usually be shown as card 1. When BCM sound is disabled, USB sound will be promoted to card 0.

For talKKonnect to know what audio devices to use (BCM2835 or USB Sound), ALSA audio config file needs to be edited. Edit file /usr/share/alsa/alsa.conf,

```
nano /usr/share/alsa/alsa.conf
```

and change

```
defaults.ctl.card 0
defaults.pcm.card 0
```

from default BCM2835 audio index (0) to the USB Sound index (1)

```
defaults.ctl.card 1
defaults.pcm.card 1
```

(This change is not necessary if BCM2835 was disabled. USB sound card will be assigned card index number “0” in that case)

USB sound device can also be set in local profile

```
nano ~/.asoundrc
```

For simple USB card cards .asound configuration like this will work:

```
  pcm.!default {
        type asym
        capture.pcm "mic"
        playback.pcm"speaker"
    }
    pcm.mic {
        type plug
        slave {
            pcm"hw:1,0"
        }
    }
    pcm.speaker {
        type plug
        slave {
            pcm"hw:1,0"
        }
    }
```

When creating .asoundrc. match the sound card index number to the exact number of the device in your system. Run ”`aplay -l`” or ”`amixer`” to check on this. You also need to match the names of capture and playback devices in this config file for your particular sound device.

Note: If the sound device was configured in global /usr/share/alsa/alsa.conf coniguration file, there is no need to create a local .asoundrc file.

Microphone or input device needs to be “captured” for talkkonnect to work. Run `alsamixer` and find your input device (mic or line in), then select it and press a space key. Red “capture” sign should show under the device in alsamixer.

Test that audio output is working by running:

```
speaker-test
```

You should hear white noise.

Test that audio input is working by looping recording to audio player:

```
arecord -f CD | aplay
```

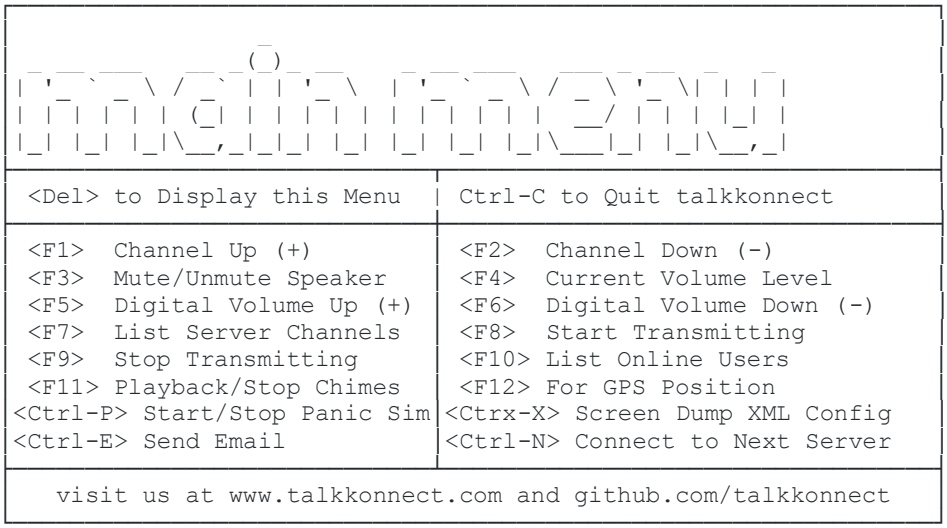
You should hear yourself speaking to the microphone.

Adjust your preferable microphone sensitivity and output gain through “alsamixer” or “amixer”, which requires some trial and error.

For a speaker muting to work when pressing a PTT, you need to enter the exact name of your audio device output in talkkonnect.xml file. This name may be different for different audio devices (e.g. Speaker, Master, Headphone, etc). Check audio output name with “aplay”, “alsamixer” or “amixer” and use that exact device name in the configuration.xml .

talKKonnect Function Keys

talKKonnect can be controled from terminal screen with function keys.



talKKonnect function keys

More information can be found at www.talkkonnect.com

suvir@talkkonnect.com