webradio, icecast, liquidsoap, mixxx, em

Radio fluflu

(page créée le 22 mars 2020)

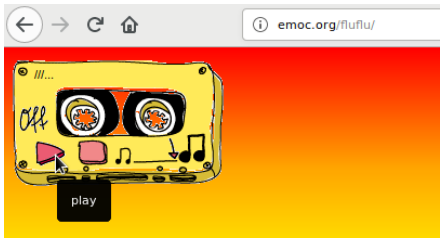
Est une webradio expérimentalo et modeste qui diffuse sur http://5.196.66.118:8000/radio

Écouter ?

On peut l'écouter en entrant l'adresse suivante dans son lecteur audio : http://5.196.66.118:8000/radio ou avec son navigateur, par une page web, quelques exemples ci-dessous.

Écouter depuis son navigateur

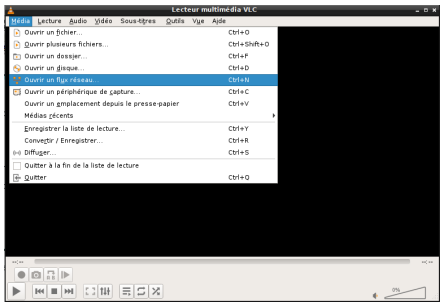
Se rendre à cette adresse (http://emoc.org/fluflu) et cliquer sur le petit bouton en triangle de la cassette...

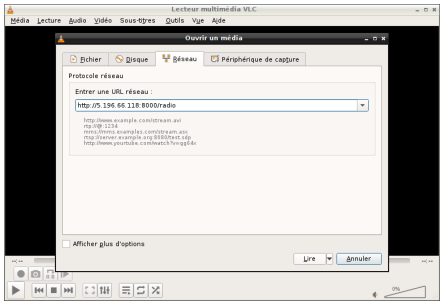


Écouter avec VLC

VLC fonctionne sur Linux, MacOS ou Windows. Il est possible de le télécharger et de l'installer depuis http://www.videolan.org/

Dans le menu “média”, choisir “ouvrir un flux réseau” et entrer l'adresse, ce qui donne en image :





Diffuser ?

La radio fonctionne en permanence à partir de playlists diffusées en fonction d'un agenda préétabli.
Il est possible de “prendre la main” pour y diffuser un livre à tout moment (le live est prioritaire sur les playlists).
Pour cela il faut envoyer un flux audio vers un point de montage (*mountpoint*).
Le fonctionnement technique est décrit plus en détail ci-dessous.

Pour envoyer ce flux, il existe une multitude de possibilités : depuis un ordi, depuis un téléphone, etc. Il sera nécessaire à chaque fois de renseigner les informations du point de montage :

- hôte (serveur) : http://5.196.66.118
- port : 8080
- point de montage : /live
- identifiant : source
- mot de passe : (contactez moi directement pour le mot de passe)
- encodage en mp3 (après avoir rencontré des problèmes avec le ogg vorbis et liquidsoap) mais le flux audio sera réencodé en OGG avant d'être diffusé.

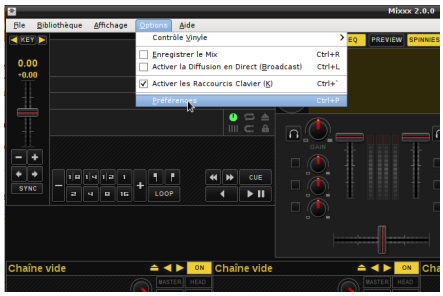
Voici quelques exemples pour diffuser :

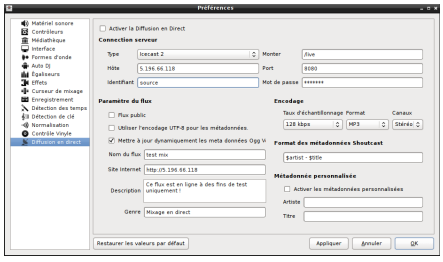
Diffuser depuis un ordinateur avec Mixxx

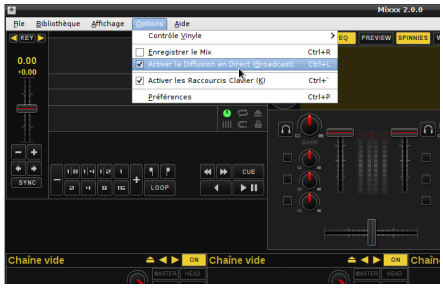
Mixxx est un logiciel libre de mixage sonore utilisable sur Linux, MacOS ou Windows. On peut le télécharger l'installer depuis https://www.mixxx.org/.
On peut l'utiliser pour mixer deux platines virtuelles, une ou plusieurs entrées son, et diffuser en direct sur un serveur, c'est ce qui nous intéresse plus particulièrement ici!

Pour diffuser avec Mixxx, il faut configurer l'accès au point de montage dans les paramètres, puis activer la diffusion en direct, tout le son qui sort de Mixxx sera transmis au serveur en temps réel (avec un léger décalage à l'écoute du à la latence de l'envoi/réception)

En image, cela donne







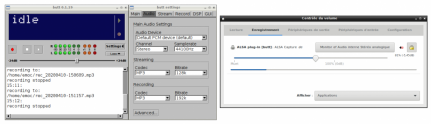
Sur l'utilisation générale de Mixxx, voir : Mixxx 2.2 User Manual

Diffuser depuis un ordinateur avec Butt

On peut utiliser butt pour streamer ce qui sort de la carte son quelque soit le logiciel utilisé

- butt http://danielnoethen.de/butt/
- https://p-node.org/documentation/streams/how-to-stream

Sur linux (debian stretch 9.5 sur un thinkpad T430), il faut compiler butt pour l'installer (c'est expliqué sur le site de l'auteur) et configurer pavucontrol comme il faut. Une petite copie d'écran pour récapituler (parce que c'était pas si simple d'arriver à ce résultat simple)



Diffuser depuis un téléphone / une tablette

(à compléter)

- (android, gratuit) broadcastmyself apk de la version gratuite https://apkpure.com/broadcastmyself/sa.broadcastmyself
- (android, libre) https://coolmic.net/ (libre, streame uniquement du ogg vorbis = prob. liquidsoap)
- (iOS, payant) http://danielnoethen.de/iziCast/

Autres possibilités

(à compléter)

- http://icecast.org/apps/

Diffuser à 2 en même temps

C'est possible aussi, il y a un 2e point de montage

- hôte (serveur) : http://5.196.66.118
- port : 8081
- point de montage : /livebis
- identifiant : source
- mot de passe : (contactez moi directement pour le mot de passe)
- encodage en mp3 (après avoir rencontré des problèmes avec le ogg vorbis et liquidsoap) mais le flux audio sera réencodé en OGG avant d'être diffusé.

Comment tout cela fonctionne t'il ?

Le serveur utilise **icecast** pour streamer les flux audio et **liquidsoap** pour gérer la planification des diffusions, et les points de montages live.

Le script liquidsoap permet de faire tourner plusieurs playlist selon l'heure de la journée et de donner la priorité aux entrées live quand elles sont utilisées

radiolive.liq

```
#!/usr/bin/liquidsoap

set("log.file.path","/tmp/radiolive.log")

# Serveur settings
set("harbor.bind_addr","0.0.0.0")

# An emergency file
emergency = single("/home/radio/test/test.mp3")

# Les playlist
emoc = mksafe(normalize(playlist(mode='random', "/home/radio/pierre")))
ortie = mksafe(normalize(playlist(mode='random', "/home/radio/ortie")))
```

```
# Playlist tournante
playlist = switch(track_sensitive=true, [ # track_sensitive true par défaut, cad attend la fin de la piste!
    ({0h-8h}, emoc),
    ({8h-10h}, ortie),
    ({10h-15h}, emoc),
    ({15h-17h}, ortie),
    ({17h-20h}, emoc),
    ({20h-22h}, ortie),
        ({22h-24h}, emoc)
    ])

# live source
live = input.harbor("live",port=8080,password="****")

#live source bis
livebis = input.harbor("livebis",port=8081,password="****")

livemix = add([live,livebis])

# fallback
radio = fallback(track_sensitive=false,
                 [livemix,playlist,emergency])

output.icecast(%vorbis,
host = "localhost", port = 8000,
password = "****", mount = "radio",
url="http://5.196.66.118:8000/radio",
name="La Baleine Radio Test",
genre="Music",
description="La Baleine Radio Test",
public=true,
fallible=true,
radio)
```

A tester

Plusieurs à envoyer des choses en même temps, ping-pong. (= plusieurs points de montage) / mixer les deux –> testé, ça fonctionne

```
#!/usr/bin/liquidsoap

set("log.file.path","/tmp/radiolive.log")

# Serveur settings
set("harbor.bind_addr","0.0.0.0")

# An emergency file
emergency = single("/home/radio/test/test.mp3")

# Les playlist
emoc = mksafe(normalize(playlist(mode='random', "/home/radio/pierre")))
ortie = mksafe(normalize(playlist(mode='random', "/home/radio/ortie")))

# Playlist tournante
playlist = switch(track_sensitive=true, [ # track_sensitive true par défaut, cad attend la fin de la piste!
    ({0h-8h}, emoc),
    ({8h-10h}, ortie),
    ({10h-15h}, emoc),
    ({15h-17h}, ortie),
    ({17h-20h}, emoc),
    ({20h-22h}, ortie),
        ({22h-24h}, emoc)
    ])

# live source
live = input.harbor("live",port=8080,password="****")

#live source bis
livebis = input.harbor("livebis",port=8081,password="****")

livemix = add([live,livebis])

# fallback
radio = fallback(track_sensitive=false,
                 [livemix,playlist,emergency])

output.icecast(%vorbis,
host = "localhost", port = 8000,
password = "****", mount = "radio",
url="http://5.196.66.118:8000/radio",
name="La Baleine Radio Test",
genre="Music",
description="La Baleine Radio Test",
public=true,
fallible=true,
radio)
```

Article extrait de : http://www.lesporteslogiques.net/wiki/ - **WIKI Les Portes Logiques**
Adresse : **http://www.lesporteslogiques.net/wiki/choses/radio_fluflu?rev=1587216414**
Article mis à jour: **2020/04/18 15:26**