

MISE EN PLACE D'UNE WEBRADIO

I- Introduction

La chaîne est constituée :

1. D'un son à diffuser (son analogique provenant par exemple du DR-2X).
2. D'une machine A équipée d'un logiciel capable d'encoder le son en flux diffusable sur internet. J'utiliserai BUTT.
3. D'un serveur diffusant ce flux sur internet. J'utiliserai IceCast qui sera hébergé sur la même machine A que BUTT. Cette machine sera donc connectée à internet. Il faudra avoir la main sur les ports.
4. D'une machine B connectée à internet équipée d'un logiciel de lecture de flux. J'utiliserai VLC

A noter que BUTT et le serveur IceCast ne sont pas nécessairement sur la même machine.

II- Installation

II.1. Sur windows (par exemple windows 10)

Télécharger IceCas depuis le site officiel : <http://icecast.org/download/> puis l'installer. Afin de simplifier les modifications du fichier de configuration icecast.xml, il convient d'installer Icecast dans un répertoire où vous avez la main. Dans mon cas, IceCast s'est installé par défaut dans C:\Program Files (x86)\Icecast où je n'ai pas les droits d'écriture... Je l'ai donc déplacé ailleurs avec un simple copier/coller ([D:\Icecast](#)).

Télécharger BUTT depuis le site officiel : <https://sourceforge.net/projects/butt/files/latest/download> puis l'installer.

II.2. Sur LINUX

II.2.a. Icecast2

Aucune difficulté :

Pour Ubuntu : icecast2 est dans les dépôts Ubuntu 16,04

Pour Debian : icecast2 se trouve à l'aide du gestionnaire de paquets Synaptic (testé Debian 9.9 avec icecast2 2.4.2)

II.2.b. Butt

Pour Butt, c'est un peu plus compliqué car il n'est pas dans le dépôt officiel.

Méthode 1 : Testée avec Ubuntu 18,04 LTS

Installer toutes les bibliothèques nécessaires au fonctionnement de BUTT

```
sudo apt-get install libfltk1.3-dev portaudio19-dev libopus-dev libmp3lame-dev  
libvorbis-dev  
libogg-dev libflac-dev libfdk-aac-dev libdbus-1-dev libsamplerate0-dev  
Butt-0.1.16 sur ubuntu16.04 installé et testé (pour autres distributions ,  
consulter le readme)
```

Télécharger le .deb sur <https://debian.pkgs.org/>

Un paquet qui a fonctionné pour est plus précisément :

https://debian.pkgs.org/10/debian-main-amd64/butt_0.1.17+dfsg-1_amd64.deb.html

Installer le paquet avec :

```
sudo dpkg --install butt_0.1.17+dfsg-1_amd64.deb
```

Remarque : il existe un paquet ...i386.deb Je pensais que ce paquet serait mieux adapté à mon ordinateur 32 bits, mais pas du tout étant donné que la version de Ubuntu est sur l'architecture amd64 (la version 32 bit n'est plus disponible au téléchargement depuis 2016).

Méthode 2 : Testée par F4HUX

On commence par installer toutes les bibliothèques nécessaires au fonctionnement de BUTT comme pour la première méthode...

On télécharge le .tar.gz sur <https://sourceforge.net/projects/butt/>

On procède à une installation manuelle.

```
cd dossier butt  
./configure  
make  
sudo make install
```

Personnellement, pour moi la compilation à l'étape mke s'est soldé par un echec (erreurs de compilation).

III- Configuration

III.1. IceCast

Pour modifier le fichier de configuration icecast.xml, j'ai utilisé sous windows « xml-copy-editor », un logiciel GNU. <https://sourceforge.net/projects/xml-copy-editor/postdownload>

Les seuls points cruciaux à configurer sont :

- le « source_password » : qui doit correspondre à celui dans BUTT
- le « admin-user »
- le « admin-password »
- le « port » utilisé,

```

<icecast>
  <!-- location and admin are two arbitrary strings that are e.g. visible
        on the server info page of the icecast web interface
        (server_version.xml).
        La Localisation est purement informative et n'a donc pas d'incidence sur le fonctionnement-->
  <location>49480 Saint-Sylvain-d'Anjou</location>
  <!-- Mail de l'admin à contacter en cas de problème, cette adresse figure sur la page du serveur, la
aussi, c'est informatif-->
  <admin>radioamateur.armi@gmail.com</admin>

  <!-- This is the hostname other people will use to connect to your server.
        It affects mainly the urls generated by Icecast for playlists and yp
        listings. You MUST configure it properly for YP listings to work!
        C'est l'adresse IP de la machine vue de l'extérieur
        La aussi c'est informatif, c'est juste l'adresse qui apparaîtra sur la page de présentation
  -->
  <hostname>Toto est arrivé</hostname>

  <!-- IMPORTANT!
        Especially for inexperienced users:
        Start out by ONLY changing all passwords and restarting Icecast.
        For detailed setup instructions please refer to the documentation.
        It's also available here: http://icecast.org/docs/
  -->

  <limits>
    <clients>50</clients>
    <sources>2</sources>
    <queue-size>524288</queue-size>
    <client-timeout>30</client-timeout>
    <header-timeout>15</header-timeout>
    <source-timeout>10</source-timeout>
    <!-- If enabled, this will provide a burst of data when a client
          first connects, thereby significantly reducing the startup
          time for listeners that do substantial buffering. However,
          it also significantly increases latency between the source
          client and listening client. For low-latency setups, you
          might want to disable this. -->
    <burst-on-connect>1</burst-on-connect>
    <!-- same as burst-on-connect, but this allows for being more
          specific on how much to burst. Most people won't need to
          change from the default 64k. Applies to all mountpoints -->
    <burst-size>65535</burst-size>
  </limits>

  <authentication>
    <!-- Sources log in with username 'source' mot de passe à renseigner dans BUTT-->
    <source-password> ??????????</source-password>
    <!-- Relays log in with username 'relay' INUTIL POUR NOTRE APPLICATION-->
    <!--relay-password>hackme</relay-password>

    <!-- Admin logs in with the username given below -->
    <!-- Il convient de ne pas laisse admin ou root comme nom administrateur par default-->
    <admin-user>F4HVG</admin-user>
    <admin-password> ??????????</admin-password>
  </authentication>

  <!-- set the mountpoint for a shoutcast source to use, the default if not
        specified is /stream but you can change it here if an alternative is
        wanted or an extension is required
  <shoutcast-mount>/live.nsv</shoutcast-mount>
  -->

  <!-- Uncomment this if you want directory listings -->
  <!--
  <directory>
    <yp-url-timeout>15</yp-url-timeout>
    <yp-url>http://dir.xiph.org/cgi-bin/yp-cgi</yp-url>
  </directory>
  -->

  <!-- You may have multiple <listener> elements -->
  <listen-socket>
    <!-- Il convient de changer le port par défaut qui est 8000 -->
    <!-- Ce port sera aussi bien utilisé par la source pour ce connecter que par les clients -->
    <port>8013</port>
    <!-- <bind-address>127.0.0.1</bind-address> -->
    <!-- <shoutcast-mount>/stream</shoutcast-mount> -->
  </listen-socket>
  <!--
  <listen-socket>
    <port>8080</port>
  </listen-socket>
  -->
  <!--
  <listen-socket>
    <port>8443</port>
    <ssl>1</ssl>
  </listen-socket>
  -->

  <!-- Global header settings
        Headers defined here will be returned for every HTTP request to Icecast.

        The ACAO header makes Icecast public content/API by default
        This will make streams easier embeddable (some HTML5 functionality needs it).
        Also it allows direct access to e.g. /status-json.xml from other sites.
        If you don't want this, comment out the following line or read up on CORS.
  -->
  <http-headers>
    <header name="Access-Control-Allow-Origin" value="*" />

```

```

</http-headers>

  <!-- Relaying
        You don't need this if you only have one server.
        Please refer to the config for a detailed explanation.
  -->
  <!--<master-server>127.0.0.1</master-server>-->
  <!--<master-server-port>8013</master-server-port>-->
  <!--<master-update-interval>120</master-update-interval>-->
  <!--<master-password>hackme</master-password>-->

  <!-- setting this makes all relays on-demand unless overridden, this is
        useful for master relays which do not have <relay> definitions here.
        The default is 0 -->
  <!--<relays-on-demand>1</relays-on-demand>-->

  <!--
  <relay>
    <server>127.0.0.1</server>
    <port>8080</port>
    <mount>/example.ogg</mount>
    <local-mount>/different.ogg</local-mount>
    <on-demand>0</on-demand>

    <relay-shoutcast-metadata>0</relay-shoutcast-metadata>
  </relay>
  -->

  <!-- Mountpoints
        Only define <mount> sections if you want to use advanced options,
        like alternative usernames or passwords
  -->

  <!-- Default settings for all mounts that don't have a specific <mount type="normal">.
  -->
  <!--
  <mount type="default">
    <public>0</public>
    <intro>/server-wide-intro.ogg</intro>
    <max-listener-duration>3600</max-listener-duration>
    <authentication type="url">
      <option name="mount_add" value="http://auth.example.org/stream_start.php"/>
    </authentication>
    <http-headers>
      <header name="foo" value="bar" />
    </http-headers>
  </mount>
  -->

  <!-- Normal mounts -->
  <!--
  <mount type="normal">
    <mount-name>/example-complex.ogg</mount-name>

    <username>othersource</username>
    <password>hackmemore</password>

    <max-listeners>1</max-listeners>
    <dump-file>/tmp/dump-example1.ogg</dump-file>
    <burst-size>65536</burst-size>
    <fallback-mount>/example2.ogg</fallback-mount>
    <fallback-override>1</fallback-override>
    <fallback-when-full>1</fallback-when-full>
    <intro>/example_intro.ogg</intro>
    <hidden>1</hidden>
    <public>1</public>
    <authentication type="htpasswd">
      <option name="filename" value="myauth"/>
      <option name="allow_duplicate_users" value="0"/>
    </authentication>
    <http-headers>
      <header name="Access-Control-Allow-Origin" value="http://webplayer.example.org" />
      <header name="baz" value="quux" />
    </http-headers>
  </mount>
  -->

  <!--
  <mount type="normal">
    <mount-name>/auth_example.ogg</mount-name>
    <authentication type="url">
      <option name="mount_add" value="http://myauthserver.net/notify_mount.php"/>
      <option name="mount_remove" value="http://myauthserver.net/notify_mount.php"/>
      <option name="listener_add" value="http://myauthserver.net/notify_listener.php"/>
      <option name="listener_remove" value="http://myauthserver.net/notify_listener.php"/>
      <option name="headers" value="x-pragma,x-token"/>
      <option name="header_prefix" value="ClientHeader:"/>
    </authentication>
  </mount>
  -->

  <fileserve>1</fileserve>

  <paths>
    <logdir>/log</logdir>
    <webroot>/web</webroot>
    <adminroot>/admin</adminroot>

    <!-- Aliases: treat requests for 'source' path as being for 'dest' path
          May be made specific to a port or bound address using the "port"

```

```

    and "bind-address" attributes.
-->
<!--
<alias source="/foo" destination="/bar"/>
-->
<!-- Aliases: can also be used for simple redirections as well,
      this example will redirect all requests for http://server:port/ to
      the status page
-->
<alias source="/" destination="/status.xml"/>
<!-- The certificate file needs to contain both public and private part.
      Both should be PEM encoded.
-->
<ssl-certificate>./icecast.pem</ssl-certificate>
-->
</paths>

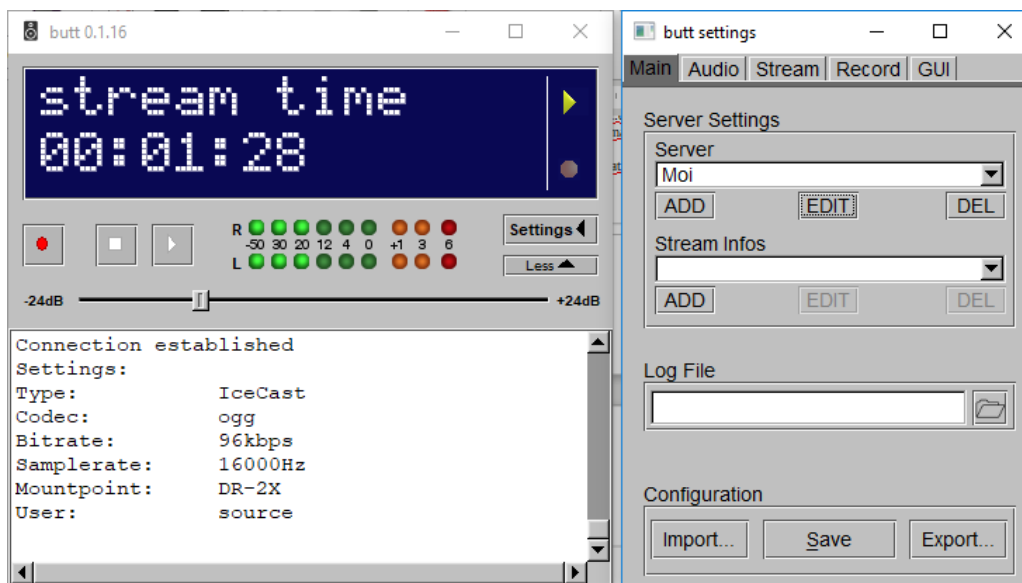
<logging>
<accesslog>access.log</accesslog>
<errorlog>error.log</errorlog>
<!-- <playlistlog>playlist.log</playlistlog> -->
<loglevel>3</loglevel> <!-- 4 Debug, 3 Info, 2 Warn, 1 Error -->
<logsize>10000</logsize> <!-- Max size of a logfile -->
<!-- If logarchive is enabled (1), then when logsize is reached
      the logfile will be moved to [error|access|playlist].log.DATESTAMP,
      otherwise it will be moved to [error|access|playlist].log.old.
      Default is non-archive mode (i.e. overwrite)
-->
<!-- <logarchive>1</logarchive> -->
</logging>
</icecast>

```

III.2. BUTT

Lancer BUTT,

III.2.a. dans l'onglet Main, ajouter un serveur.



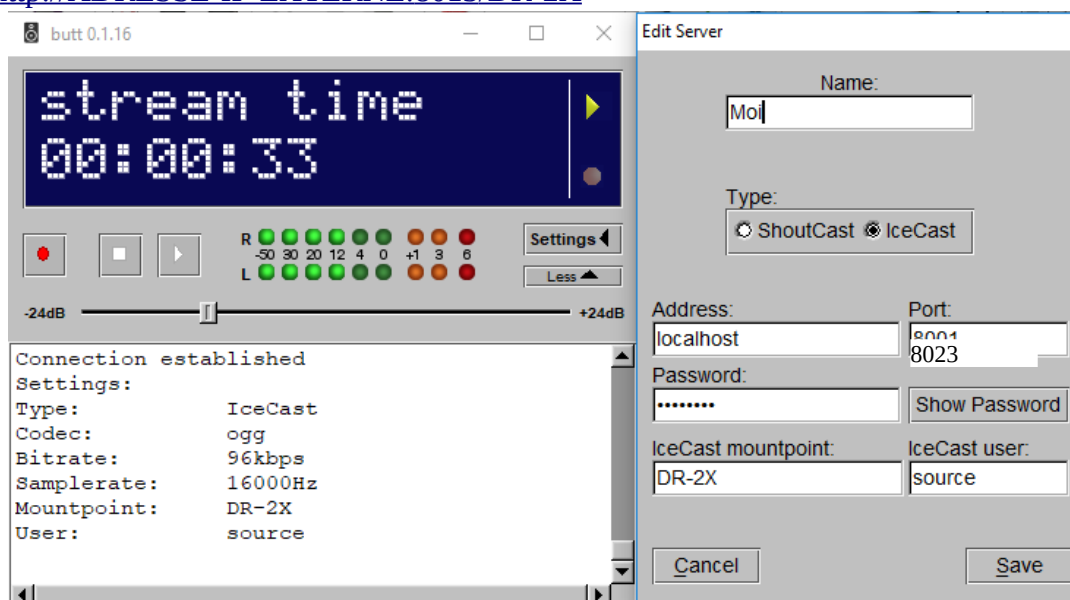
Il s'agit d'ajouter notre serveur IceCast. Le nom « Moi » n'a aucune importance.

Cliquer sur ADD ou EDIT pour définir ou modifier les paramètres d'un serveur déjà configuré.

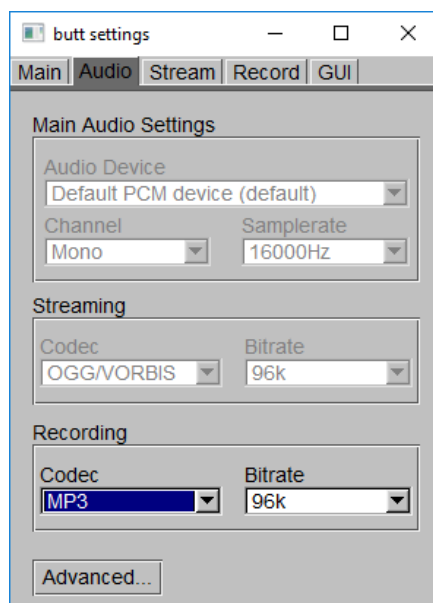
L'adresse pourra rester en « localhost » tant que c'est la même machine qui héberge BUTT et IceCast. **Le port 8013 est celui que l'on a renseigné dans icecast.xml. Le mot de passe est celui défini par source-password dans icecast.xml.** IceCast user est laissé par défaut « source ».

Icecast mountpoint permettra de lire le flux sur internet avec un lien url du type :

<http://ADRESSE-IP-EXTERNE:8013/DR-2X>



III.2.b. Onglet Audio



Tous les paramètres que j'ai pu tester fonctionnent correctement.

La seule problème que j'ai rencontré est l'absence de choix dans « Audio Device ». Il faut alors aller dans les paramètres audios de Windows et **configurer le périphérique d'entrée**.

III.2.c. Autres onglets

Stream (sert uniquement pour une vraie WebRadio) et GUI (interface graphique) sont laissés par défaut.

Record permet d'enregistrer le flux dans un fichier : fonction testée avec succès.

III.3. VLC

Sur la machine B. Après avoir installé VLC.

Faire : Média/Ouvrir un flux réseau

Dans Entrer une URL réseau : <http://ADRESSE-IP-EXTERNE:8013/DR-2X>

D'autres lecteurs existent à commencer par votre navigateur favori.

Rappel : port 8013 configuré dans icecast.xml et DR-2X défini dans BUTT (IceCast Mountpoint).

IV- Applications

On peut se demander à quoi un tel peut bien servir dans le cadre du DR-2X !

1. A savoir si je suis relayé sans avoir le QSL d'un autre OM
2. A savoir si ma station est meilleure en émission ou en réception (on peut accrocher le relay sans entendre le retour par radio).
3. A savoir jusqu'à quel niveau de puissance je suis entendu. En effet, il ne sert à rien de pousser ses ampli dans leurs limites...
4. A écouter le QSO sans être à sa station mais depuis n'importe où dans le monde dès que l'on a une connexion internet.
5. A promouvoir les activités du club en mettant un lien sur le site de l'ARML pour que les personnes intéressées par la radio puissent se mettre à l'écoute.

6. Un enregistrement du flux audio peut permettre de mesurer l'utilisation du relay et d'identifier d'éventuels piratages.

D'autres applications sont à inventer en dehors du relai...

V- Conclusion

La technologie employée étant initialement prévue pour la webradio, elle engendre un décalage de quelques secondes. Ce qui a paru au départ être un inconvénient ouvre en réalité à de nouvelles applications.

Pour supprimer le décalage entre la captation du flux audio et sa disponibilité sur internet, il faudrait utiliser le principe de le VO-IP (Voice over Internet Protocol).

VI- Liens consultés :

<https://www.le85.com/forum/2-le-forum-de-rivendell-automation-le85-com/9-astuce-envoi-de-flux-audio-le-streaming-facile-sous-ubuntu-xubuntu-avec-butt.html>
http://wiki.dane.ac-versailles.fr/index.php?title=Installer_OS_webradio
<https://philippe.scoffoni.net/radio-web-logiciels-libres/>

Autres liens non utilisés qui offrent des solutions semblant prometteuses.

<https://blog.microlinux.fr/webradio-mpd-icecast-1/>

Par F4HVG le 12/12/2018, modifié par F4HUX (partie Linux)

Mise à jour le 05/04/2019, le 15/05/2019