

**BBS Packet Radio**

**Winlink**

**Raspberry Pi /Raspbian**



## Table des matières

1. Direwolf.....	4
Dépendances :.....	4
Compilation :.....	4
Configuration :.....	5
2. Outils ax25.....	8
Installation.....	8
Configuration.....	8
3. Etablissement d'une Liaison BBS.....	9
Etape 1 : Direwolf.....	9
Etape 2 : axcall.....	10
4. PAT - Winlink.....	11
Installation.....	11
Configuration.....	11
Utilisation.....	13
5. Annexe.....	15
HAMLIB.....	15
Installation.....	15
Configuration.....	16
Références.....	17

Le but de ce document est de vous aider à mettre en place les outils nécessaire afin de pouvoir communiquer depuis un Raspberry Pi vers un serveur BBS ou encore une passerelle Winlink via Radio.

Votre Raspberry Pi devra impérativement être connecté à Internet pendant toute la phase d'installation, cependant à l'issue de l'installation il sera parfaitement fonctionnel sans aucun accès à Internet.

Le matériel utilisé pour la validation de ce document :

- Raspbian Pi 3 sous Raspbian (version avril 2019)
- Interface carte son Yaesu SCU-17
- Yaesu FT-857D

Dans le suite de la documentation, Direwolf jouera le rôle de TNC logiciel, les outils ax25 seront utilisés pour se connecter à un BBS, et PAT pour dialoguer avec un Node Winlink.

Pour chacun des composants, il est recommandé d'utiliser un terminal différent, ceci est facilement réalisable lors de l'utilisation de Raspbian Desktop. Si vous utilisez le mode console je vous conseille d'utiliser tmux, votre moteur de recherche préféré vous dira tout sur tmux.

Si vous avez déjà installé les outils précédemment, rendez-vous directement au Chapitre 3 : Etablissement d'une Liaison BBS.

# 1. Direwolf

## Dépendances :

```
sudo apt-get install libasound2-dev git
```

Récupérer la dernière version de Direwolf :

```
git clone https://www.github.com/wb2osz/direwolf
```

## Compilation :

```
cd direwolf
```

```
make
```

```
sudo make install
```

```
make install-conf
```

```
make install-rpi
```

## Configuration :

Avant de commencer l'édition de la configuration il faut récupérer les informations concernant l'interface carte son.

```
aplay -l
```

```
pi@raspberrypi:~/direwolf $ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: ALSA [bcm2835 ALSA], device 0: bcm2835 ALSA [bcm2835 ALSA]
  Subdevices: 7/7
    Subdevice #0: subdevice #0
    Subdevice #1: subdevice #1
    Subdevice #2: subdevice #2
    Subdevice #3: subdevice #3
    Subdevice #4: subdevice #4
    Subdevice #5: subdevice #5
    Subdevice #6: subdevice #6
card 0: ALSA [bcm2835 ALSA], device 1: bcm2835 ALSA [bcm2835 IEC958/HDMI]
  Subdevices: 1/1
    Subdevice #0: subdevice #0
card 1: CODEC [USB Audio CODEC], device 0: USB Audio [USB Audio]
  Subdevices: 0/1
    Subdevice #0: subdevice #0
```

```
arecord -l
```

```
pi@raspberrypi:~/direwolf $ arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: CODEC [USB Audio CODEC], device 0: USB Audio [USB Audio]
  Subdevices: 0/1
    Subdevice #0: subdevice #0
```

L'interface carte son étant une Yaesu SCU-17, elle se nomme [USB Audio CODEC], donc Card 1 et Device 0.

Editer direwolf.conf :

**nano direwolf.conf**

```
#####  
#                                                                 #  
#           FIRST AUDIO DEVICE PROPERTIES                       #  
#           (Channel 0 + 1 if in stereo)                         #  
#                                                                 #  
#####  
  
#  
# Many people will simply use the default sound device.  
# Some might want to use an alternative device by choosing it here.  
#  
# Linux ALSA is complicated. See User Guide for discussion.  
# To use something other than the default, generally use plughw  
# and a card number reported by "arecord -l" command. Example:  
  
ADEVICE plughw:1,0
```

Dé-commenter ADEVICE et indiquer les références de la carte son 1,0 (Card,Device) trouvée à l'étape précédente.

Renseigner votre indicatif au niveau du paramètre « MYCALL » en remplaçant « NOCALL »

```
MYCALL NOCALL
```

Renseigner les paramètres pour CAT :

```
# COM1 can be used instead of /dev/ttyS0, COM2 for /dev/ttyS1, and so on.  
#  
  
#PTT COM1 RTS  
#PTT COM1 RTS -DTR  
PTT /dev/ttyUSB1 RTS DTR  
  
#
```

Les paramètres ci-dessus conviennent pour l'interface SCU-17.

La commande « `dmesg | grep tty` » peut vous aider à identifier le port USB de votre équipement.

exemple :

```
pi@raspberrypi:~/direwolf $ dmesg | grep tty
[ 0.000000] Kernel command line: 8250.nr_uarts=0 bcm2708_fb.fbwidth=656 bcm2708_fb.fbheight=416 bcm2708_fb.fbswap=1 vc_mem.mem_base=0x3ec00000 vc_mem.mem_size=0x40000000 dwc_otg.lpm_enable=0 console=ttyS0,115200 console=tty1 root=PARTUUID=f5d121ac-02 rootfstype=ext4 elevator=deadline fsck.repair=yes rootwait quiet splash plymouth.ignore-serial-consoles
[ 0.000284] console [tty1] enabled
[ 0.664999] 3f201000.serial: ttyAMA0 at MMIO 0x3f201000 (irq = 87, base_baud = 0) is a PL011 rev2
[ 72.632131] usb 1-1.5.1: cp210x converter now attached to ttyUSB0
[ 72.639270] usb 1-1.5.1: cp210x converter now attached to ttyUSB1
```

Sachant que cp210x correspond au SCU-17, le port USB est le ttyUSB0 ou ttyUSB1, il faudra tester pour déterminer le meilleur choix.

Enregistrer le fichier de configuration.

Pour démarrer Direwolf, dans un terminal dédié:

```
./direwolf -t 0 -p
```

Pour quitter Direwolf : CTRL+C

## 2. Outils ax25

### Installation

```
sudo apt-get install libax25 ax25-apps ax25-tools
```

### Configuration

Éditer le fichier de configuration des ports ax25 :

```
sudo nano /etc/ax25/axports
```

Ajouter une ligne en utilisant votre indicatif

```
GNU nano 2.9.3 /etc/ax25/axports
# /etc/ax25/axports
#
# The format of this file is:
#
# name callsign speed paclen window description
#
#1 OH2BNS-1 1200 255 2 144.675 MHz (1200 bps)
#2 OH2BNS-9 38400 255 7 TNOS/Linux (38400 bps)
1 F4HU0-7 1200 255 2 BBS
```

Retenez bien le premier chiffre de la nouvelle ligne, le « 1 », il correspond au numéro de port AX25, nous en aurons besoins par la suite.

Vous pouvez mettre autre chose que « 1 » si ça vous arrange, même un nom, il faut juste s'en rappeler par la suite...



### 3. Etablissement d'une Liaison BBS

#### Etape 1 : Direwolf

Ouvrir un terminal et rendez-vous dans le répertoire direwolf.

Démarrer direwolf avec la commande :

```
./direwolf -t 0 -p
```

Un écran similaire doit apparaître :

```
pi@raspberrypi:~/direwolf $ ./direwolf -t 0 -p
Dire Wolf version 1.5

Reading config file direwolf.conf
Audio device for both receive and transmit: plughw:1,0 (channel 0)
Channel 0: 1200 baud, AFSK 1200 & 2200 Hz, E+, 44100 sample rate / 3.
Ready to accept AGW client application 0 on port 8000 ...
Ready to accept KISS TCP client application 0 on port 8001 ...
Virtual KISS TNC is available on /dev/pts/3
Created symlink /tmp/kisstnc -> /dev/pts/3
█
```

Notez bien le port du KISS TNC : `/dev/pts/3`

Dans un autre terminal, lancer la commande :

```
sudo /usr/sbin/kissattach /dev/pts/3 1
```

`/dev/pts/3` vient de l'étape précédente, le « `1` » correspond au numéro de port ax25.

```
pi@raspberrypi:~ $ sudo /usr/sbin/kissattach /dev/pts/3 1
AX.25 port 1 bound to device ax0
```

## Etape 2 : axcall

Pour contacter un BBS, depuis un terminal, utiliser la commande axcall suivi du QRZ du BBS à contacter :

```
axcall 1 QRZ
```

(remplacer QRZ avec l'indicatif du Node)

En cas d'utilisation d'un relais BBS

```
axcall 1 QRZ_1 via QRZ_2
```

(comme d'habitude, le 1 correspond au numéro/nom du port ax25)

Pour écouter les messages packet qui passent, il est soit possible de retourner dans le terminal direwolf ou utiliser la commande axlisten.

```
sudo axlisten
```

## 4. PAT - Winlink

PAT est un client Winlink, il peut être utilisé en ligne de commande ou via une interface graphique. Nous ne nous intéresserons qu'à l'interface graphique dans ce document.

Pour que PAT puisse prendre en charge le pilotage de votre TX, il est nécessaire d'installer HAMLIB. L'installation de HAMLIB est disponible en annexe de ce document.

### Installation

```
wget https://github.com/la5nta/pat/releases/download/v0.6.1/pat_0.6.1_linux_armhf.deb
```

Rendez-vous sur le site <https://getpat.io> pour avoir la dernière version.

```
sudo dpkg -i pat_0.6.1_linux_armhf.deb
```

### Configuration

Pour éditer la configuration, lancer la commande suivante :

```
pat configure
```

Dans le fichier qui va s'ouvrir remplacer les valeurs MyCall, MyPassword, MyLocator (MyPassword correspond à votre mot de passe Winlink) avec vos informations personnelles.

```
{
  "mycall": "MyCALL",
  "secure_login_password": "MyPassword",
  "auxiliary_addresses": [],
  "locator": "MyLocator",
  "http_addr": "0.0.0.0:8080",
  "motd": [
    "Open source Winlink client - getpat.io"
  ],
}
```

Vous pouvez en profiter pour remplacer dans http\_addr , localhost:8080 avec 0.0.0.0:8080 afin de rendre l'interface graphique accessible depuis n'importe quel équipement de votre réseau, sinon il ne sera accessible que depuis la machine locale.

La suite de la configuration est à faire uniquement si la prise en charge de CAT via HAMLIB est nécessaire.

Mettre à jour la section « hamlib\_rigs ». Pour le nom du TX (FT-857), le texte est libre, il faut cependant utiliser le même dans le reste de la configuration.

```
"FT857": {"address": "localhost:4532", "network": "tcp"}
```

Mettre à jour la section « ax25 »

```
{
  "mycall": "MyCALL",
  "secure_login_password": "MyPassword",
  "auxiliary_addresses": [],
  "locator": "MyLocator",
  "http_addr": "0.0.0.0:8080",
  "motd": [
    "Open source Winlink client - getpat.io"
  ],
  "connect_aliases": {
    "telnet": "telnet://{mycall}:CMSTelnet@cms.winlink.org:8772/wl2k"
  },
  "listen": [],
  "hamlib_rigs": {
    "FT857": {"address": "localhost:4532", "network": "tcp"}
  },
  "ax25": {
    "port": "1",
    "beacon": {
      "every": 3600,
      "message": "Winlink P2P",
      "destination": "IDENT"
    },
    "rig": "FT857",
    "ptt_ctrl": true
  },
}
```

"port" : "1", = Numéro du port ax25 configuré dans le chapitre ax25

"rig" : "FT857" = Nom du TX défini précédemment

"ptt\_ctrl" : true = Activation (true) de la commande PTT via CAT

Enregistrer la configuration en faisant Ctrl-O et quitter avec Ctrl-X

## Utilisation

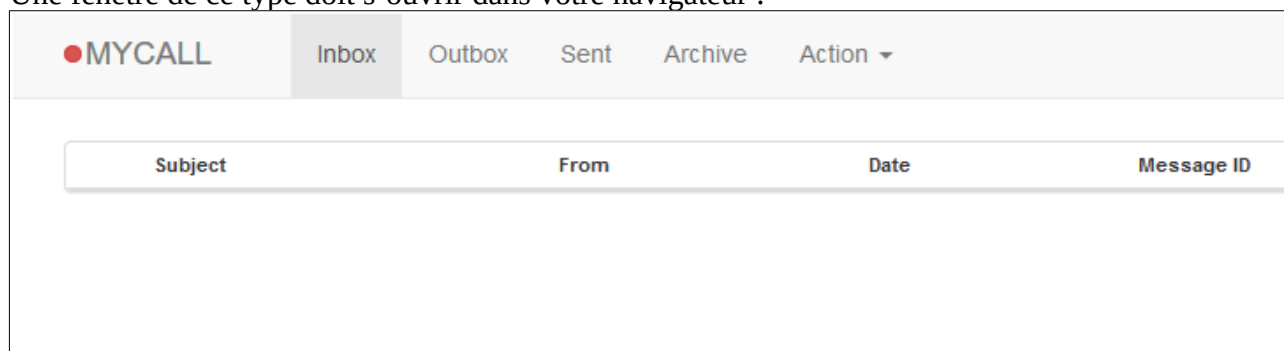
Avant de pouvoir utiliser PAT, démarrer Direwolf en suivant l'Etape 1 du Chapitre 3.

Démarrer PAT avec la commande

```
pat http
```

Ouvrir un navigateur internet sur votre Pi et accéder à l'URL : <http://localhost:8080>

Une fenêtre de ce type doit s'ouvrir dans votre navigateur :

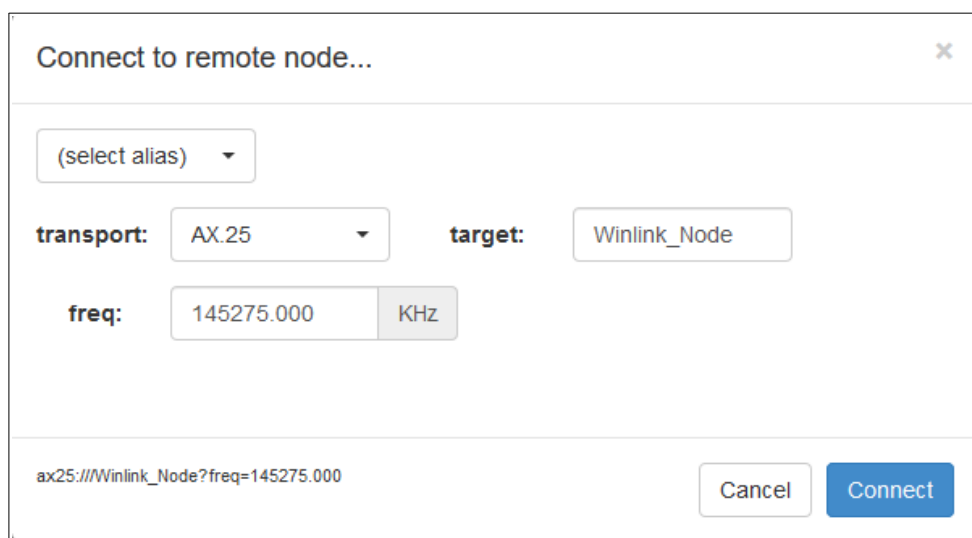


The screenshot shows a web interface for 'MYCALL'. At the top, there is a navigation bar with 'Inbox' selected, and other tabs for 'Outbox', 'Sent', 'Archive', and an 'Action' dropdown. Below the navigation bar is a table with the following columns: 'Subject', 'From', 'Date', and 'Message ID'. The table is currently empty.

Le menu Action vous permettra de rédiger des messages et de vous connecter à un serveur Winlink.

Allez dans le menu Action puis Connect.

Afin d'initier une connexion radio, sélectionner comme transport ax25, dans target indiquer l'indicatif du Node Winlink à contacter et enfin la fréquence en KHz (nécessaire pour le pilotage CAT) puis cliquer sur Connect.

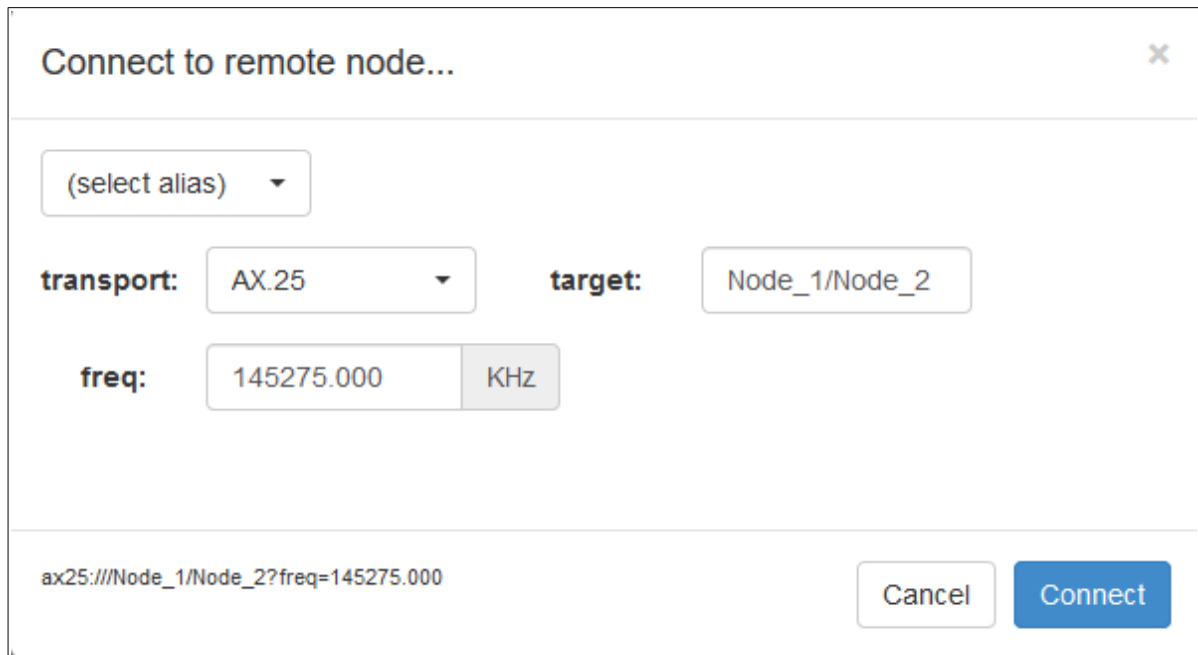


The screenshot shows a dialog box titled 'Connect to remote node...'. It contains the following fields and controls:

- A dropdown menu labeled '(select alias)'.
- A 'transport:' field with a dropdown menu set to 'AX.25'.
- A 'target:' field with a text input containing 'Winlink\_Node'.
- A 'freq:' field with a text input containing '145275.000' and a unit dropdown set to 'KHz'.
- A text area at the bottom showing the generated URL: 'ax25:///Winlink\_Node?freq=145275.000'.
- 'Cancel' and 'Connect' buttons at the bottom right.

Au bout de quelques secondes la connexion devrait s'établir, et PAT va récupérer/envoyer les messages.

Pour établir une connexion via un digipeater, il faut l'indiquer dans le champ target en les séparant d'un /

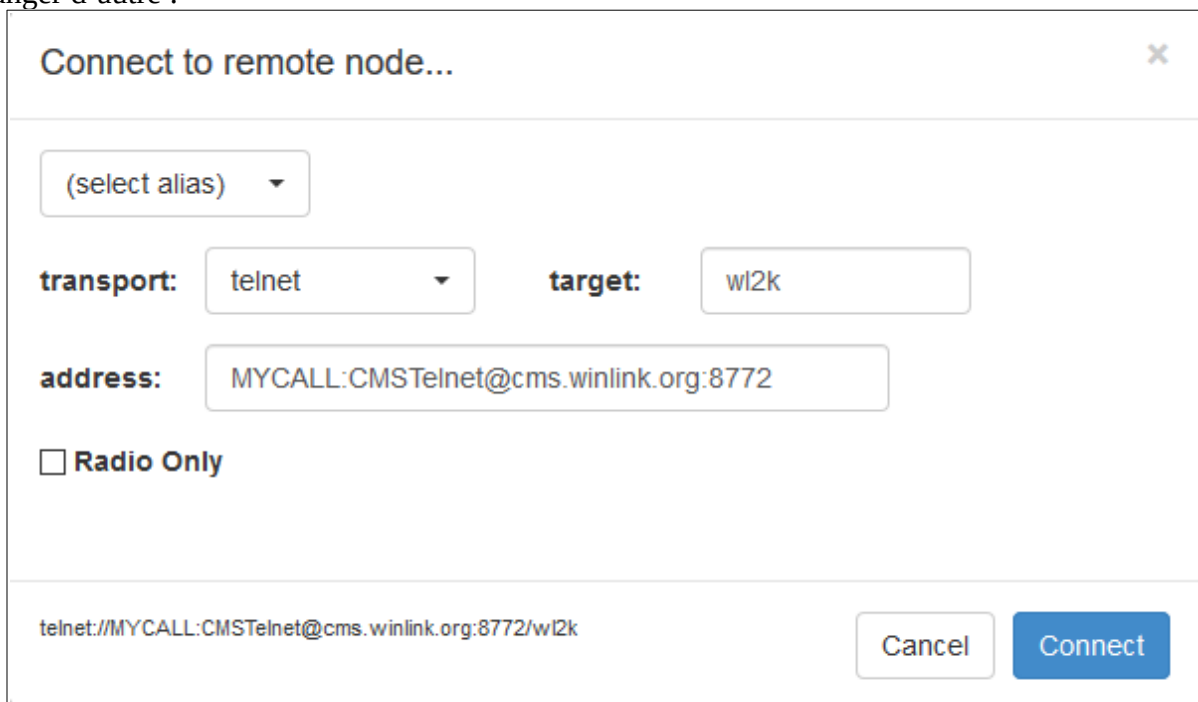


The dialog box titled "Connect to remote node..." contains the following fields and controls:

- A dropdown menu labeled "(select alias)" with a downward arrow.
- A "transport:" label followed by a dropdown menu showing "AX.25".
- A "target:" label followed by a text input field containing "Node\_1/Node\_2".
- A "freq:" label followed by a text input field containing "145275.000" and a unit dropdown menu showing "KHz".
- A preview field at the bottom left showing the constructed URL: "ax25:///Node\_1/Node\_2?freq=145275.000".
- "Cancel" and "Connect" buttons at the bottom right.

Node\_1 = digipeater / Node\_2 = Node Winlink

Pour établir une connexion via internet, sélectionner telnet dans le premier ascenseur et ne rien changer d'autre :



The dialog box titled "Connect to remote node..." contains the following fields and controls:

- A dropdown menu labeled "(select alias)" with a downward arrow.
- A "transport:" label followed by a dropdown menu showing "telnet".
- A "target:" label followed by a text input field containing "wl2k".
- An "address:" label followed by a text input field containing "MYCALL:CMSTelnet@cms.winlink.org:8772".
- A checkbox labeled "Radio Only" which is currently unchecked.
- A preview field at the bottom left showing the constructed URL: "telnet://MYCALL:CMSTelnet@cms.winlink.org:8772/wl2k".
- "Cancel" and "Connect" buttons at the bottom right.

## 5. Annexe

### HAMLIB

HAMLIB est une suite d'outils permettant de piloter votre radio via CAT

#### Installation

```
sudo apt-get install libhamlib-utils
```

Lancer la commande ci-dessous pour lister les appareils supportés

```
rigctl --list
```

```
pi@raspberrypi:~ $ rigctl --list
```

Rig #	Mfg	Model	Version	Status
1	Hamlib	Dummy	0.5	Beta
2	Hamlib	NET rigctl	0.3	Beta
101	Yaesu	FT-847	0.5	Beta
103	Yaesu	FT-1000D	0.0.6	Alpha
104	Yaesu	MARK-V FT-1000MP	0.0.5	Alpha
105	Yaesu	FT-747GX	0.4.1	Beta
106	Yaesu	FT-757GX	0.4.1	Beta
107	Yaesu	FT-757GXII	0.4	Stable
109	Yaesu	FT-767GX	1.0	Stable
110	Yaesu	FT-736R	0.3	Stable
111	Yaesu	FT-840	0.1	Untested
113	Yaesu	FT-900	0.1	Untested
114	Yaesu	FT-920	2010-08-23	Stable
115	Yaesu	FT-890	0.1	Stable
116	Yaesu	FT-990	0.2.1	Alpha
117	Yaesu	FRG-100	0.4	Beta
118	Yaesu	FRG-9600	0.2	Untested
119	Yaesu	FRG-8800	0.2	Untested
120	Yaesu	FT-817	0.5.1	Beta
121	Yaesu	FT-100	0.4.1	Beta
122	Yaesu	FT-857	0.5	Beta
123	Yaesu	FT-897	0.3.3	Beta

Notez dans la première colonne le numéro correspondant à votre radio.

La radio utilisée dans ce document étant le Yaesu FT-857, son numéro est le 122.

## Configuration

Il n'y a pas de configuration particulière, pour démarrer la prise en charge de CAT via HAMLIB, lancer dans un terminal séparé la commande suivante :

```
rigctld -m 122 -r /dev/ttyUSB0 -s 4800
```

122 correspondant au numéro de votre TX, USB0 le port CAT et 4800 la vitesse du port série.

A l'exécution de cette commande rien ne se passe, réduisez simplement la fenêtre.



## Références

<https://www.kevinhooke.com/2015/09/12/direwolf-soundcard-packet-on-linux-with-ax25-and-linpac/>

<https://www.kevinhooke.com/2019/02/22/packet-radio-on-debian-9-with-direwolf-and-ax25/>

<http://linpac.sourceforge.net/screenshots.php>

<https://github.com/Hamlib/Hamlib>

<http://getpat.io/>