

# LoRaWAN Gateway Kit

Chaigne Hyacinthe

2e semestre 2021

## Contents

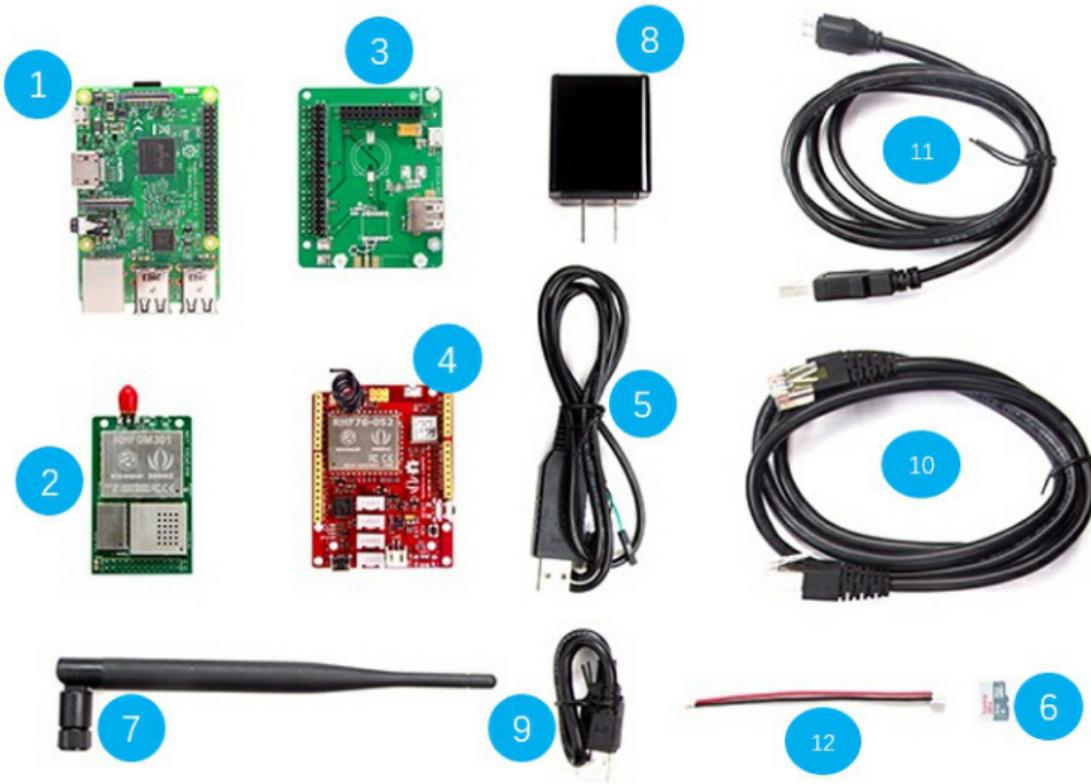
<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Contenu</b>	<b>2</b>
<b>3</b>	<b>Assemblage</b>	<b>3</b>
<b>4</b>	<b>Logiciels</b>	<b>4</b>
<b>5</b>	<b>Mise en place</b>	<b>5</b>
5.1	Passerelle (Gateway) . . . . .	5
5.2	Node (Noeud) . . . . .	9
<b>6</b>	<b>Accès au serveur</b>	<b>11</b>
6.1	ABP mode . . . . .	12
6.2	OTAA mode . . . . .	12
<b>7</b>	<b>Loriot serveur</b>	<b>13</b>
7.1	Passerelle (Gateway) . . . . .	14
7.2	Node (Noeud) . . . . .	15
<b>8</b>	<b>Ressources</b>	<b>16</b>



## 1 Introduction

Maintenant que vous êtes sensibilisés à la technologie LoRaWAN nous vous proposons ici de fabriquer une passerelle, un nœud et un serveur local. Le but est ici de s'initier grâce à un environnement opérationnel aux simples moyens de paramétrage. La force de ce kit réside dans le bas coût et les possibilités de développement/analyses d'un réseau LoRaWAN. Il est modulaire, fiable et peu consommateur en ressources. A noter que même si son prix est plus élevé que les produits déjà prêts, ce kit nous permet des libertés supplémentaires. [Lien du kit](#).

## 2 Contenu



N	Nom
1	Raspberry Pi 3B
2	Module passerelle RHF0M301-868
3	PRI 2 Bridge RHF4T002
4	Seeeduino LoRaWAN avec GPS (RHF76-052AM)
5	Adapteur USB vers UART
6	Extension Micro SD Card 16GO – Classe 10
7	0dBi Rubber Duck Antenna
8	5V/2.1A Adapteur standard américain avec connecteur Micro USB
9	Câble micro USB 20cm
10	Micro USB Cable 100cm
11	Câble RJ45 Ethernet 200cm
12	Câble JST2.0 10cm

Le kit s'assemblera autour de la Raspberry Pi (un mini ordinateur) où nous y grefferons le module LoRa (la couche physique) afin d'exploiter la technologie de communication (LoRaWAN) avec une Arduino (carte programmable) embarquant ledit module LoRa. Le bridge sert simplement d'intermédiaire entre la Raspberry et le module LoRa.

### 3 Assemblage



Tout comme indiqué sur ces deux images, commencez par assembler la Raspberry (**1**) avec le bridge (**3**), ceux ci fermement fixés ajouté le module LoRa au dessus (**2**). Branchez le micro USB de la couche inférieure avec l'USB de la couche supérieure avec le micro câble (**9**). Accessoirement vous pouvez visser l'antenne (**7**) sur son socle couleur or.

Voici en pratique à quoi doit ressembler votre gateway assemblée. Notez que l'alimentation se fait sur le bridge (**3**).



## 4 Logiciels

Vous aurez besoin d'un outil pour communiquer avec votre gateway, est un juste milieu car il possède une possibilité de lecture via Serial port et SSH. Vous pouvez bien entendu utiliser un autre logiciel. Dans un second temps vous aurez besoin d'Arduino et la bibliothèque de la carte (seeeduino LoRaWAN).

### Linux

- Sur Ubuntu :

```
$ sudo add-apt-repository universe
$ sudo apt update
$ sudo apt install putty
$ sudo apt install arduino
```

- Sur Arch :

```
$ sudo pacman -S putty ou $ yay putty
$ sudo pacman -S arduino ou $ yay arduino
```

- Sur CentOS, Red Hat Fedora :

```
$ sudo yum install putty ou $ sudo dnf install putty
$ sudo yum install arduino ou $ sudo dnf install arduino
```

- Depuis le source code :

```
PuTTY : lien
$ tar -xvf putty-0.73.tar.gz $ cd putty-0.73/
$ ./configure
$ sudo make sudo make install
```

Arduino : [lien](#)

### Windows7+

- PuTTY : [lien](#)
- Arduino : [lien](#)

**Mac** Nous préconisons de plutôt établir une machine virtuelle Windows car peu de solutions aux problèmes sont recensés sur les forums.

- PuTTY :

- Via Homebrew : \$ brew install putty
- Via MacPorts : \$ sudo port install putty  
et \$ cp /opt/local/bin/putty ~/Desktop/PuTTY pour ajouter un raccourci.

- Arduino : [lien](#)

## 5 Mise en place

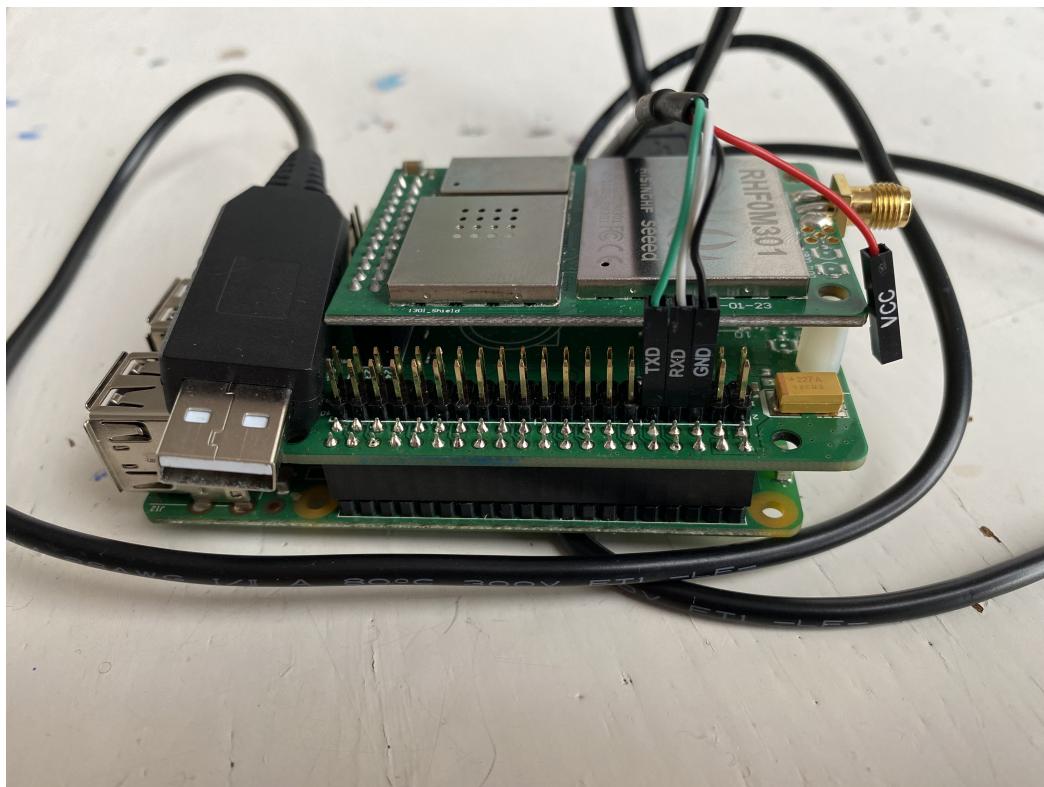
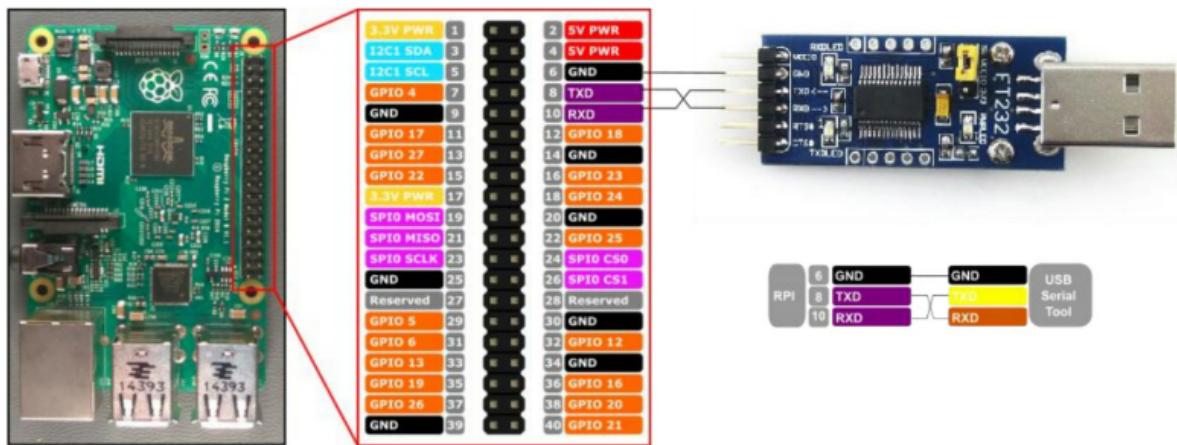
Avant d'établir un quelconque contact entre la gateway et le noeud, il vous faut dans un premier temps les paramétrier. Nous vous proposons deux approches, choisissez celle qui vous paraît la plus pratique pour vous.

### 5.1 Passerelle (Gateway)

#### Approche 1 - Serial port

Insérez la micro USB dans le port au dos de la Raspberry, celle-ci contient un OS (Raspberry Pi OS) adéquat. Cet OS est spécialisé pour le micro ordinateur et offre des performances optimales. Notons que les services que nous utiliserons y sont déjà prés-installés. Au cas où celle-ci venait à devenir défaillante, faites une sauvegarde système de l'OS.

Branchez maintenant le cable sur les pins selon le schéma suivant tel que :



Avant de mettre la Raspberry sous tension, branchez le câble USB à votre ordinateur. Vous devez trouver sur quel port celle-ci est affectée avant de pouvoir utiliser votre lecteur de Serial port.

- **Linux**

Installez **setserial** par exemple puis :

```
$ sudo setserial -g /dev/tty*
```

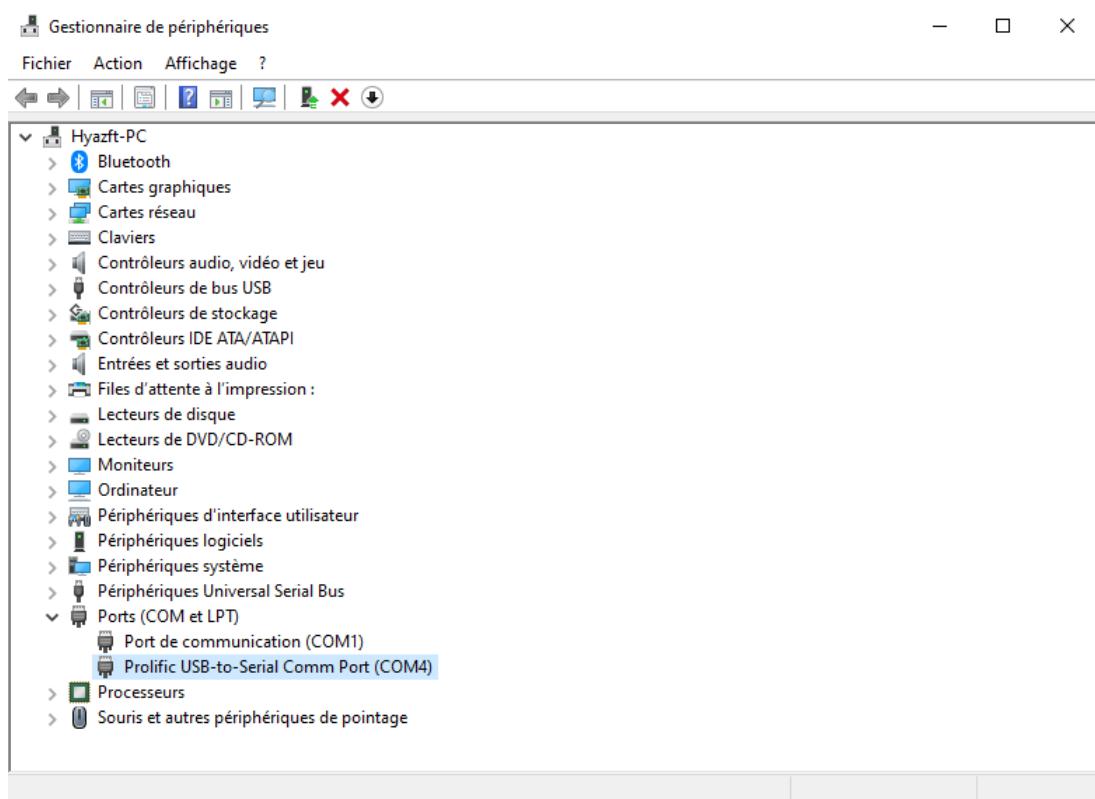
Vous devriez avoir un résultat similaire :

```
/dev/ttyS0, UART: unknown, Port: 0x03f8, IRQ: 4
/dev/ttyS1, UART: unknown, Port: 0x02f8, IRQ: 3
/dev/ttyS2, UART: unknown, Port: 0x03e8, IRQ: 4
/dev/ttyS3, UART: unknown, Port: 0x02e8, IRQ: 3
/dev/ttyUSB0, UART: 16654, Port: 0x0000, IRQ: 0
```

Vous déduirez qu'ici nous utilisons le port **/dev/ttyUSB0**.

- **Windows7+**

Naviguez jusqu'au gestionnaire des périphériques via le lanceur Windows en bas à gauche.

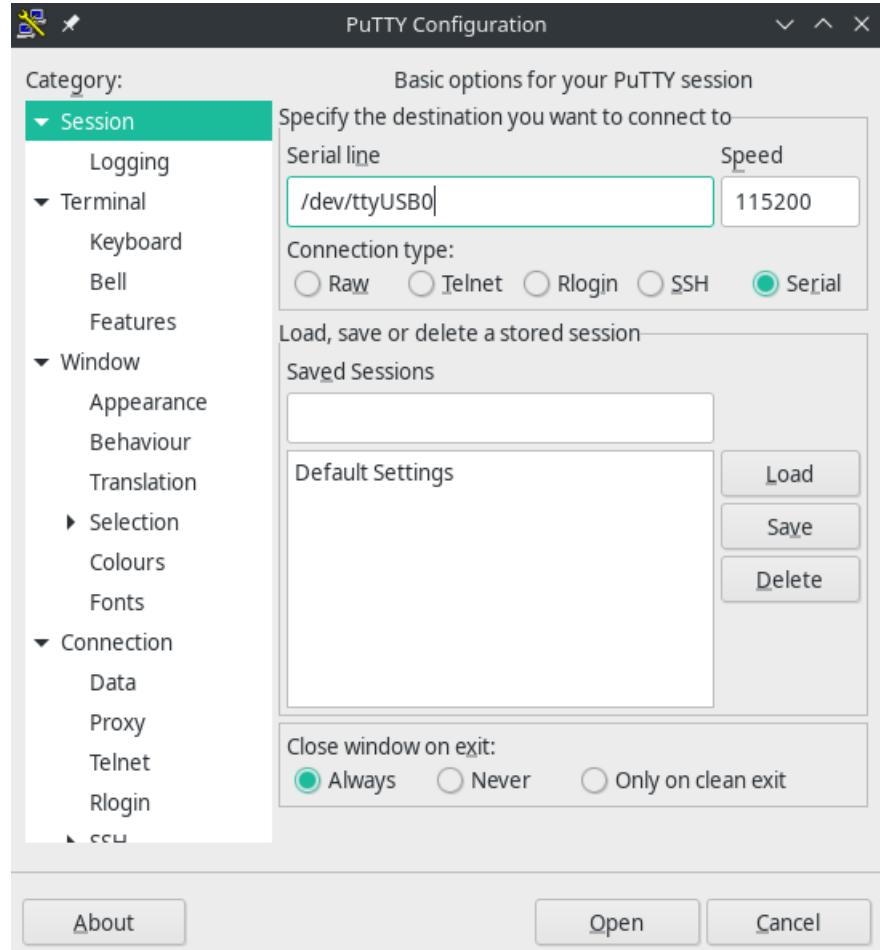


Il s'agira dans le cas présent du port **COM4**.

- **Mac**

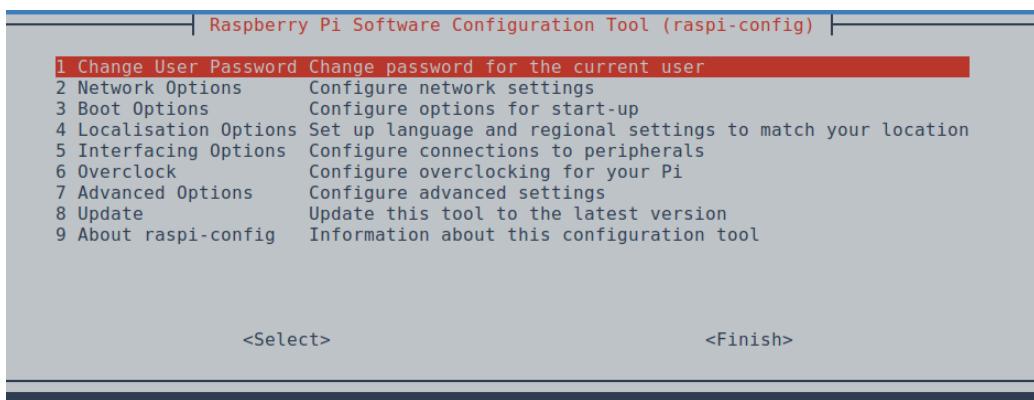
Il n'est pas trop tard pour acheter un ordinateur. Tentez cependant `$ ls /dev/tty.*` ou vérifier vos ports dans le menu hardware mac. Bon courage.

Lancez maintenant PuTTY et sélectionnez **Serial**, insérez le port que vous avez trouvé dans **Serial line** et la vitesse (bauds) à **115200**. Cliquez ensuite sur **open**. Vous devriez voir une console vide.



En dernière étape, alimentez le bridge (3), après quelques secondes (ou jusqu'à 2 mins) vous devriez voir apparaître des données, c'est la Raspberry qui boot et vous communique sa console. ID : rxhf, PASS : risinghf ou berrylora.

Bonus : Si vous souhaitez par la suite travailler en SSH, reliez le câble Ethernet ou éditez **wpa\_supplicant** comme dans l'approche 2 si vous préférez le wifi. Autorisez enfin le SSH dans les configurations via la commande **sudo raspi-config**. Naviguez dans "Interfacing options" et activez le SSH.



Sous le même réseau que votre Raspberry, fermez la connexion existante et sélectionnez l'onglet SSH. Dans la partie adresse notez y : **rhf2s001.local**. Si vous avez déjà une console sous la main : **ssh rxhf@rhf2s001.local**.

### Approche 2 - Direct SSH

Le SSH étant nativement désactivé de base, retirez la micro USB et ouvrez ses fichiers via un ordinateur. Accédez à la partition **boot** et créez un fichier vide nommé **ssh** sans extension.

Si vous préférez le wifi au cablage ethernet, editez le fichier **/etc/wpa\_supplicant/wpa\_supplicant.conf** en y ajoutant :

```
network={  
    ssid="wifi_ssid"  
    psk="wifi_password"  
}
```

Insérez enfin la micro USB dans la Raspberry et mettez la sous tension. Dans PuTTY sélectionnez l'onglet SSH. Dans la partie adresse notez y : **rhf2s001.local**. Si vous avez déjà une console sous la main : **ssh rxhf@rhf2s001.local**.

### TO DO

Maintenant que vos accès préférés sont paramétrés vous pouvez commencer à communiquer avec la Raspberry et le module. Commencez par mettre à jour le système :

```
$ sudo apt update sudo apt upgrade -y
```

Maintenant vérifiez que le serveur LoraWAN fonctionne :

```
$ sudo systemctl status pktfwd
```

**Si le status est inactif :**

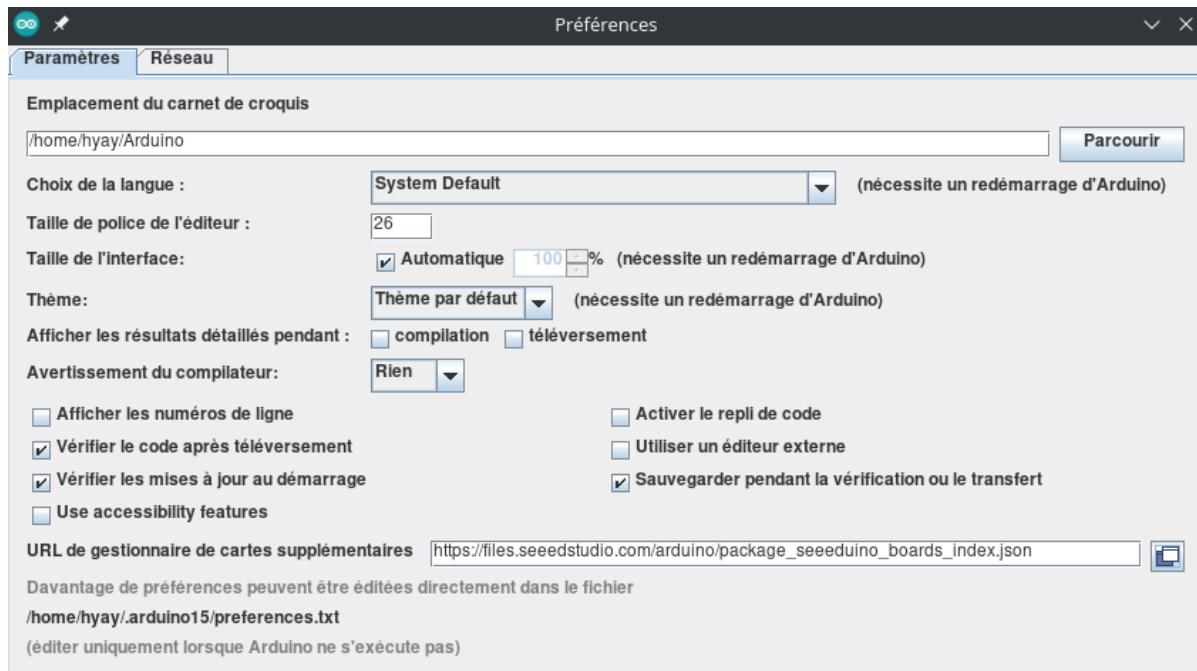
<pre>\$ sudo systemctl enable pktfwd</pre>	#active le service au démarrage
<pre>\$ sudo systemctl restart pktfwd</pre>	#(Re)lance le service

	EU868	Uplink DR
CH0	867.1	DR0 ~ DR5
CH1	867.3	DR0 ~ DR5
CH2	867.5	DR0 ~ DR5
CH3	867.7	DR0 ~ DR5
CH4	867.9	DR0 ~ DR5
CH5	868.1	DR0 ~ DR5
CH6	868.3	DR0 ~ DR5
CH7	868.5	DR0 ~ DR5

Parce que vous êtes sous législation européenne, ce kit utilise le plan EU868. Vous aurez donc 8 bandes paramétrées comme ceci. Votre serveur LoRa est désormais actif.

## 5.2 Node (Noeud)

Tout en laissant votre serveur sous tension, munissez vous de la carte Seeeduino LoRaWAN et lancez le logiciel Arduino. Nous aurons besoin d'ajouter les drivers de la carte. Pour ce faire naviguez dans Fichiers puis Préférences et ajoutez l'url [https://files.seeedstudio.com/arduino/package\\_seeeduino\\_boards\\_index.json](https://files.seeedstudio.com/arduino/package_seeeduino_boards_index.json) dans le champs concerné. Si vous y avez déjà ajouté des lignes, faites simplement un retour à la ligne avant de l'insérer.



Le lien désormais renseigné, naviguez dans le gestionnaire de cartes dans l'onglet Outils et installez les drivers **Seeed SAMD Boards**. S'il n'apparaît pas, assurez vous que l'url que vous avez rentré est correct. Prenez soin de bien renseigner le port (détecté automatiquement) dans l'onglet Outils.

```
#include <SoftwareSerial.h>

void setup() {
    Serial1.begin(9600);
    SerialUSB.begin(115200);
}

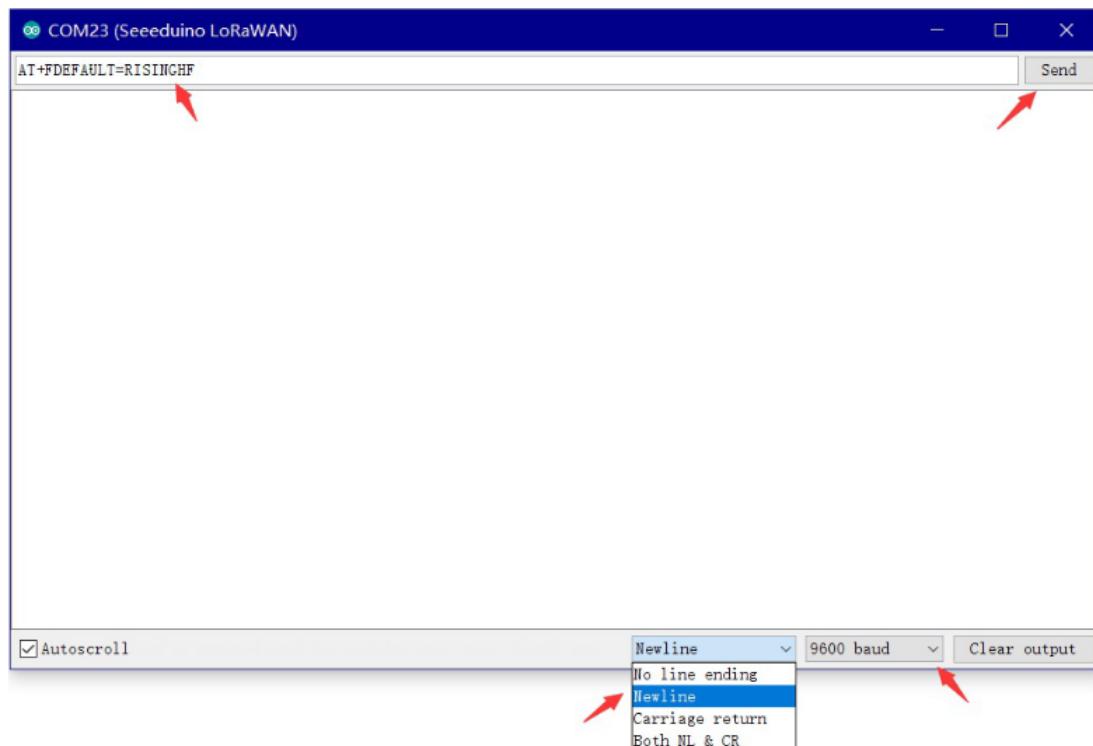
void loop() {
    while(Serial1.available()) {
        SerialUSB.write(Serial1.read());
    }

    while(SerialUSB.available()) {
        Serial1.write(SerialUSB.read());
    }
}
```

Sélectionnez ensuite la carte dans Outils, Type de carte, Seeeduino\_LoRAWAN. Compilez puis téléversez le programme. Si vous rencontrez des soucis sur Linux, il s'agit sûrement des permissions, référez vous à la [documentation ici](#).

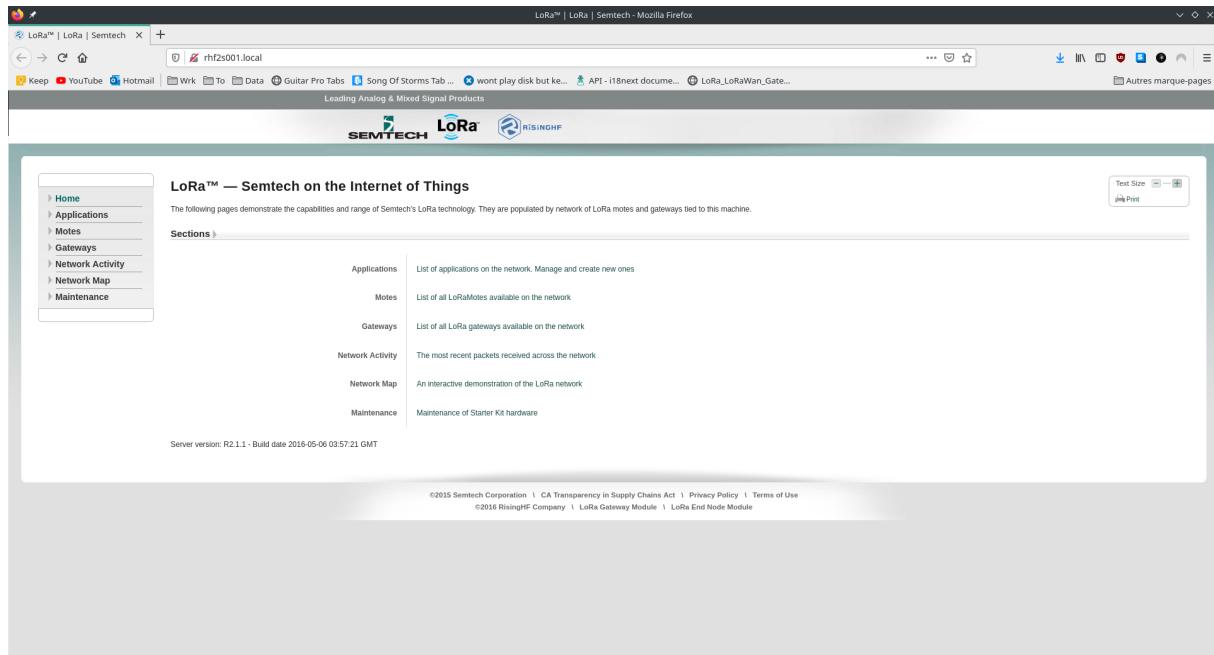
Enfin ouvrez le moniteur de série (la loupe en haut à droite), cochez **nouvelle ligne** et insérez les lignes suivantes :

```
AT+FDEFAULT=RISINGHF
AT+DR=EU86
```



## 6 Accès au serveur

Accédez à la page du serveur en indiquant l'adresse IP ou le nom de domaine **rhf2s001.local**. Il est plus facile de gérer ici qu'en lignes de commandes.



Vous avez, dans un premier temps, besoin de lier le noeud à sa passerelle. Pour ce faire nous devons renseigner les identifiant du noeud sur cette page. Dans le moniteur de série tapez **at+id**, vous obtiendrez les identifiant de votre carte seeeduino LoRaWAN.

- DevEUI : Identifiant unique (64bits) (adresse physique)
- DevAddr : Adresse 32bits dynamique
- AppEUI : Identifiant unique (64bits) recensé sur The Things Network

The Things Network est un réseau recensant des passerelles publiques. Nous pouvons à partir d'ici axer le projet de deux manières. OTAA (Over-the-Air Activation) ou ABP (Activation by Personalization). OTAA (payant) est la manière la plus sécurisée pour se connecter au TTN, chaque puce doit passer par une procédure de jointure durant laquelle DevAddr se voit assigner une adresse et ses clefs de sécurité. ABP (gratuit) offre lui une manière de hard coder DevAddr et les clefs (donc manuellement). Il est logique que cette partie soit en théorie moins sécurisée. Plus de détails sur [ce lien](#).

Vous devriez voir apparaître un résultat similaire :

```
+ID: DevAddr, 00:81:12:6D
+ID: DevEui, 47:DB:55:80:00:29:00:1F
+ID: AppEui, 52:69:73:69:6E:67:48:46
```

## 6.1 ABP mode

Les appareils peuvent communiquer avec les applications auxquelles ils ont été enregistrés. Pour enregistrer un appareil, vous devez d'abord ajouter une application. Cloquez donc sur l'onglet correspondant. Renseignez un nom, un propriétaire et **l'AppEUI**. Rendez vous dans votre application et dans Configure motes. C'est ici que vous attacherez des appareils à votre application. Renseignez les informations dans la partie Personalised Motes comme suit :

- DevEui AT+ID command
- DevAddr: AT+ID command
- NWKSKEYDefault value 2B7E151628AED2A6ABF7158809CF4F3C
- APPSKEYDefault value 2B7E151628AED2A6ABF7158809CF4F3C

Vous pouvez par la suite tester la communication.

at+mode=lwabp AT+CMSGHEX="0a 0b 0c 0d 0e"	# Spécifie utilisation mode ABP # Envoi paquet
--	---

```
+ID: DevAddr, 00:81:12:6D
+ID: DevEui, 47:DB:55:80:00:29:00:1F
+ID: AppEui, 52:69:73:69:6E:67:48:46
+MODE: LWABP
+CMSGHEX: Start
+CMSGHEX: Wait ACK
+CMSGHEX: ACK Received
+CMSGHEX: RXWIN1, RSSI -47, SNR 8.3
+CMSGHEX: Done
```

## 6.2 OTAA mode

Créez une application avec **l'AppEui** : **00-00-00-00-00-00-00-01**. Retournez dans le moniteur série Arduino et changez l'adresse AppEui : **at+id=appeui,"00-00-00-00-00-00-00-01"**.

Nouveau résultat accessible via : **at+id**.

Rendez vous dans la partie Configure motes de votre application et renseignez le **DevEUI** et **AppKey**. Cliquez sur ajouter et retournez dans le moniteur série :

at+mode=lwotaa at+join AT+CMSGHEX="0a 0b 0c 0d 0e"	# Spécifie utilisation mode OTAA # Fait la jointure avec TTN # Envoi paquet
--	---

Mote (DevEUI)	Application Key (AppKey)
New: 47:97:c5:34:00:1d:00:48	2B7E151628AED2A6ABF7158809CF4F3C
<input type="button" value="Add"/>	

```

at+mode=lwotaa
+MODE: LWOTAA

at+join
+JOIN: Starting
+JOIN: NORMAL, count 1, 0s, 0s
+JOIN: Network joined
+JOIN: NetID 000000 DevAddr 00:82:2c:96
+JOIN: Done

AT+CMSGHEX="0a 0b 0c 0d 0e"
+CMSGHEX: Start LoRaWAN transaction
+CMSGHEX: TX "0A 0B 0C 0D 0E "
+CMSGHEX: Wait ACK
+CMSGHEX: ACK Received
+CMSGHEX: RXWIN1, RSSI -47, SNR 3.25
+CMSGHEX: Done

```

## 7 Loriot serveur

Il est possible de manager ses passerelles et ses noeuds depuis une plateforme en ligne. Elle facilite le paramétrage dans la mesure où tous les appareils sont disponibles recensés sur le suite sont par la suite accessibles où l'on veut. Ces appareils seront attachés à un **compte**. Commencez donc par en créer un [ici](#). Vous devriez après en pratique obtenir une page similaire à celle ci-dessous.

The screenshot shows the Loriot Community Account interface. On the left, a sidebar lists navigation options: Dashboard, Applications (1), Networks (1), Join Servers, Documentation, Account, Upgrade, and Support. The main content area has several sections:

- COMMUNITY ACCOUNT**: Welcome message for joining the Loriot ecosystem.
- News**: Two news items: one from April 21, 2020, about the release of Loriot Network Server 6.0, and another from February 20, 2020, encouraging users to upgrade to Professional Public Server.
- Gateways of Sample network**: A table showing a single gateway entry:
 

Location	Model	MAC	Version	Last Data
FR Lille	Raspberry Pi 3	BB-27-EB-FF-C9-DF-14 BB27EBFFC9DF14	2.8.1341-JKS-EU1-25	never
- Applications**: A table showing a single application entry:
 

Name	AppID	Devices
SampleApp	BE-7A-24-E6	1

Avant de pouvoir utiliser cette solution il faut installer les dépendances sur la Raspberry et recenser notre noeud. Notez que vous pouvez utiliser cette méthode plus graphique en addition ou simplement à la place de la méthode manuelle de départ.

## 7.1 Passerelle (Gateway)

La notion de "Network" sur le site correspond à une collection de passerelle, ceci à des fins pratiques de gestion. Rendez vous dans ladite rubrique et créez un Network. Puis ajoutez un Gateway à ce Network en cliquant sur **Register a new Gateway**. Prenez soin de sélectionner votre **Raspberry 3B** qui apparaît dans la liste. Indiquez dans le menu déroulant :

- Radio front-end : RHF2S001 868/915 MHz(SX1257)
- BUS : SPI

Notez plus bas l'adresse MAC de votre passerelle via la commande `$ ifconfig eth0 | grep HWaddr`. Vous verrez désormais sous le Network que vous avez créé la Gateway inactive. En cliquant sur celle-ci assurez vous que le banc de fréquence est bien réglé sur 868Mhz comme ici.

The screenshot shows the configuration interface for a gateway. At the top, there are three tabs: 'Community Access' (marked with a green checkmark), 'Alert Notifications' (marked with a red X), and 'Ignore Data' (marked with a red X). Below these tabs, the 'Region' is set to 'EU863-870'. Under the 'Antennas' section, there is a table with columns 'ID', 'Tx Gain', and 'Channel Plans'. A single row is present with ID 0, Tx Gain 0, and Channel Plan 'EU868\_Semtech' (which is highlighted in blue). At the bottom right of the table, there are buttons for '+ Add Band' and '- Remove Plans'.

Toujours dans le même Network et la même Gateway, naviguez dans l'onglet **Software** situé sur votre gauche, il s'agit des instructions d'installation du serveur Loriot. Les étapes d'installation peuvent varier, il est recommandé de suivre la page. Au cas où voici les étapes actuelle d'installation sur la Raspberry :

```

$ sudo apt-get install ca-certificates
$ sudo apt-get install libftdi1

Installation de LORIOT Gateway self-extracting installer (et pas les binaries !)
$ cd /tmp
$ wget https://eu1.loriot.io/home/gwsw/loriot-pi-3-rhf1257-SPI-0-latest.sh -O loriot-install.sh
$ chmod +x loriot-install.sh
$ ./loriot-install.sh -f -s eu1.loriot.io

```

Après ces commandes le serveur tourne et se lancera automatiquement au démarrage. Vous devriez voir le serveur comme "Active" quelques minutes plus tard.

## 7.2 Node (Noeud)

Vous pouvez, dans l'onglet **Radio**, voir les fréquences disponibles.

EU868_Semtech		
Frequency	Bandwidth	Modulation
867.100 MHz	125 kHz	MultiSF
867.300 MHz	125 kHz	MultiSF
867.500 MHz	125 kHz	MultiSF
867.700 MHz	125 kHz	MultiSF
867.900 MHz	125 kHz	MultiSF
868.100 MHz	125 kHz	MultiSF
868.300 MHz	250 kHz	SF7
868.300 MHz	125 kHz	MultiSF
868.500 MHz	125 kHz	MultiSF
868.800 MHz	125 kHz	FSK

Retournez à votre moniteur de série Arduino et modifiez les bandes de fréquences comme ceci :

Liste des fréquences utilisées par l'Arduino :  
at+ch

Modifier les fréquences utilisées :  
at+ch=0,868.1  
at+ch=1,868.3  
at+ch=2,868.5

Tout comme manuellement, vous devez créer une application avant de pouvoir enregistrer une puce en ABP ou OTAA. Revenez donc à la page de menu du site et naviguez dans l'onglet **Application** puis **Enroll device**. Faites attention à bien choisir la version du LoRaWAN utilisé par votre puce sur l'Arduino (regardez la documentation). Et rentrez les informations comme vu précédemment dans la partie manuelle. Pour rappel :

DevAddr: Commande "AT+ID"  
FCntUpPar defaut : 1  
FCntDnPar defaut : 1  
NWKSKEYPar defaut : 2B7E151628AED2A6ABF7158809CF4F3C  
APPSKEYPar defaut : 2B7E151628AED2A6ABF7158809CF4F3C  
EUIDEVEUI, Commande "AT+ID"

Vous pouvez par la suite tester la communication.

at+mode=lwabp	# Spécifie utilisation mode ABP
AT+CMSGHEX="0a 0b 0c 0d 0e"	# Envoi paquet

Naviguez ensuite dans votre application et la partie **Devices**, votre noeud s'y trouve. En cliquant dessus vous obtenez sa liste d'activité. Vous devriez voir apparaître le message envoyé dans **Last data**. De même plus généralement sur la page de votre application.

## 8 Ressources

- Lien kit <https://www.seeedstudio.com/LoRa-LoRaWAN-Gateway-868MHz-Kit-with-Raspberry-Pi-3.html>
- Site loriot <https://loriot.io/>
- Documentation Loriot <https://docs.loriot.io/display/LNS/Getting+Started>