

Table des matières

Introduction	2
1 Architecture du réseau LoRaWAN	2
2 Installation des différents éléments	3
2.1 Gateway	4
2.1.1 Liste de matériel	4
2.1.2 Préparation du matériel	4
2.1.3 Semtech packet forwarder	6
2.2 Chirpstack Gateway Bridge	7
2.3 Chirpstack Network Server	8
2.4 Chirpstack Application Server	9
Conclusion	12

Introduction

La technologie LoRa (Long Range) sert à la communication d'objets connectés. Grâce à des signaux émis à bas débit, transmis par ondes radio à faible consommation électrique, mais de longue portée, les objets IoT dotés de son protocole réseau LoRaWAN (long range radio wide area network) sont connectés à internet par des passerelles. Dans la suite de ce document nous allons decire la procédure d'installation des différents composants présents dans un reseau LoRaWAN.

1 Architecture du réseau LoRaWAN

Le réseau LoRaWAN est composés des ces différents éléments que nous pouvons voir sur le schéma ci-dessous.

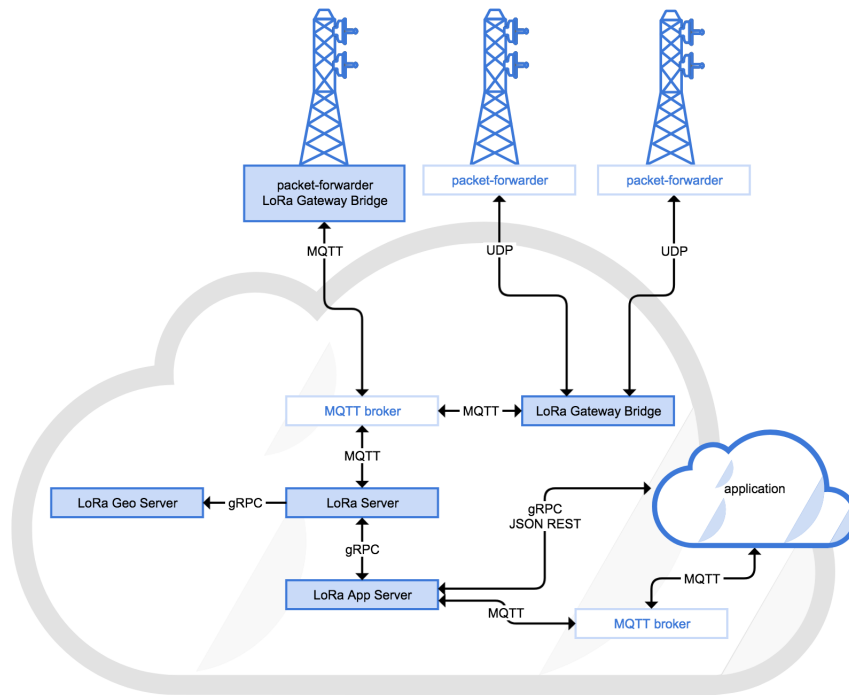


FIGURE 1 – Architecture de LoRaWAN

Un *noeud* est un point communiquant avec le Gateway et où des mesures sont relevées.

Le *Gateway*, passerelle en français, est un point pouvant communiquer avec les noeuds. Il achemine les données envoyées par les noeuds vers le Gateway-Bridge.

Le *Gateway-Bridge* permet de convertir le format des données renvoyées par le Gateway dans des formats de données exploitables par le Network-Server [2].

Le *Network-Server* a comme principal rôle la deduplication des messages reçus par les gateways : dans le cas où l'on dispose de plusieurs Gateway et que le message envoyé par un noeud a été capté par au moins deux de ces Gateways, il sauvegarde celui du Gateway le mieux disposé et l'achemine vers l'Application-Server. Visitez le site en [5] pour plus d'information.

L'*Application-Server* est l'interface d'utilisateur. C'est à ce niveau l'on observe les données envoyées par les noeuds et c'est également là qu'est faite toute configuration permettant la communication avec les noeuds [8].

Notons que c'est le protocole TCP/IP qui est utilisé pour la communication entre les éléments suivants : le Gateway, le Gateway-Bridge, le Network-Server et l'Application-Server. Entre le Gateway et les noeuds, c'est LoRa qui est utilisé pour la communication. La communication entre un noeud et l'Application-Server est bidirectionnelle ; ce qui permet au noeud d'envoyer de message à l'Application-Server et d'en recevoir de ce dernier.

2 Installation des différents éléments

Les éléments peuvent être installés sur de différentes machines comme on peut les installer sur une même machine (le Gateway dans ce cas). La procédure d'installation décrite ici est celle dont les éléments sont installés sur une même machine. La procédure ne prend pas en compte le noeud. Les composants de LoRa-Gateway-Bridge, LoRa-Network-Server et LoRa-Application-Server sont des anciennes versions. Respectivement, les nouvelles versions de ces programmes sont chirpstack-Gateway-Bridge, chirpstack-Network-Server et chirpstack-Application-Server.

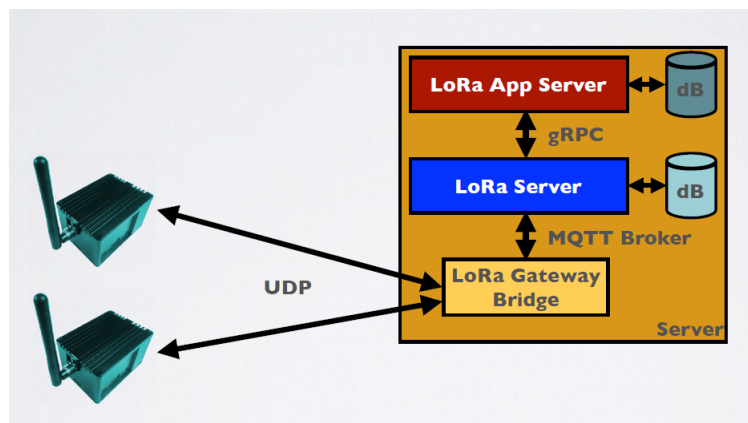


FIGURE 2 – Les éléments constituant un LoRaWAN

2.1 Gateway

Le Gateway est un ensemble constitué du matériel d'un concentrateur (RAK831) et d'un hôte (Raspberry). Un hôte étant un nano-ordinateur sur lequel s'exécute un programme appelé *packet-forwarder*. Parfois un module GPS peut s'ajouter au concentrateur et à l'hôte pour former le Gateway.

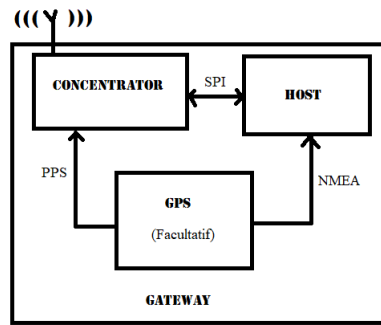


FIGURE 3 – Gateway et ses composants [18]

2.1.1 Liste de matériel

La liste du matériel utilisé pour notre Gateway est la suivante :

- un Raspberry Pi 3 model B+
- un module RAK831

2.1.2 Préparation du matériel

Pour commencer, il faut monter le module RAK831 sur le Raspberry. Le document en [15] et la vidéo en [21] montre comment le montage se faire. Dans la suite, nous décrirons comment le Raspberry doit être configuré.

1. Installation de l'OS (Raspbian) sur le Raspberry

Allez sur :

- le lien en [12] pour télécharger la dernière version du Raspbian.
- le lien en [11] pour télécharger Elcher (logiciel qui va permettre de flasher la carte SD du Raspberry).

Une fois que ces deux fichiers sont téléchargés :

- Placez la carte SD dans un adaptateur puis la brancher à votre PC
- Formatez la carte SD
- Lancez le logiciel Elcher
- Sélectionnez le fichier compressé du Raspbian puis la carte SD
- Cliquer sur démarrer pour lancer l'opération de flashing de la carte SD

2. Accès au Raspberry

Une fois l'opération terminée, créez sur la partition de démarrage les deux fichiers suivants :

- Un fichier nommé `wpa_supplicant.conf` et contenant le texte ci-dessous.

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=BJ
network={
    ssid="ssid"
    psk="mot-de-passe"
}
```

Ceci permettra au Raspberry de se connecter au réseau Wi-Fi ayant le ssid en question [13].

- Un fichier sans extension et sans contenu nommé **ssh**. Ceci permettra d'avoir accès au Terminal du Raspberry quand ce dernier sera connecter au même réseau que votre PC.
- Placez la carte SD dans le Raspberry, ensuite branchez l'antenne du RAK831 avant d'alimenter le Raspberry (recommandation du fabricant RAK).
- Une fois le Raspberry démarré, allez dans le Terminal (Linux/MacOS) de votre PC puis tapez la commande :

```
>> ssh pi@addr_ip_du_raspberry
```

tapez le mot de passe : **raspberrry**

Ainsi vous avez accès au Terminal du Raspberry (connexion-ssh).

- Activez l'interface SSH.
Le fichier **ssh** créé permet la connexion-ssh après le démarrage du Raspberry et ensuite il est supprimé. Pour éviter d'enlever la carte SD dans le but de créer de nouveau le fichier *ssh* à chaque fois, il faut activer le SSH permanamment. Pour le faire, accédez au menu de configuration du Raspberry en tapant dans le Terminal la commande suivante :

```
>> sudo raspi-config
```

Ensuite sélectionnez dans le menu de configuration le l'option **Interfacing Option** puis l'option **SSH** puis **activer**

3. Les opérations à effectuer

- Activez l'interface SPI.
Pour le faire, sélectionnez dans le menu l'option **Interfacing Option** puis l'option **SPI** puis **activer**
- Changez le fuseau horaire (timezone) en celui de votre pays.
Pour le faire, sélectionnez dans le menu de configuration l'option **Localisation Option** puis l'option **Change Timezone** puis choisissez le continent et la ville capitale de votre pays.
- Pour des raisons de sécurité changez le mot de passe par défaut de l'utilisateur **pi**. Sélectionnez l'option **Change User Password** pour changer le mot de passe.
- Une fois les modifications effectuées, quittez le menu de configuration

et le raspberry se redémarrera. Reconnectez vous et tapez la commande suivante pour mettre à jour le OS.

```
>> sudo apt-get update && sudo apt-get upgrade -y
```

2.1.3 Semtech packet forwarder

Le tout ne suffit pas de relier les différents pour avoir un gateway. Il faut aussi un programme nommé Semtech packet forwarder qui sera exécuté sur l'hôte.

- Une fois la mise à jour terminée, tapez la commande suivante pour installer git.

```
>> sudo apt-get install git
```

- Lorsque l'installation de git sera terminée, nous allons passer au téléchargement du Semtech-Packet-Forwarder. Notons que la procédure normale d'installation est décrite en [19] ; mais Robert Lie a simplifié cette procédure en créant un fichier nommé `install.sh` qu'on peut retrouver avec le lien [20]. Téléchargez Semtech-packet-forwarder en tapant :

```
>> git clone https://github.com/robertlie/RAK831-LoRaGateway-RPi ~/rak831-loragateway
```

- Accédez au repertoire du dossier téléchargé

```
>> cd ~/rak831-loragateway
```

- Lancez l'installation en exécutant le script suivant :

```
>> sudo ./install.sh
```

Les lignes suivantes se dérouleront. Saisissez les informations correspondantes ou tapez sur la touche **Entrer** pour garder les valeurs par défaut.

```
Host name [ttn-gateway]: <Enter>
Region AS1, AS2, AU, CN, EU, IN, KR, RU, US [EU]: EU
Latitude [0]: <Enter>
Longitude [0]: <Enter>
Altitude [0]: <Enter>
```

NB : Il faut noter que ce script exécute installe *Semtech LoRa library* dans le répertoire `/opt/ttn-gateway/lora_gateway` et *Semtech UDP Packet Forwarder* dans le répertoire `/opt/ttn-gateway/packet_forwarder`

- Accédez et éditez le fichier de configuration global du packet forwarder.

```
>> sudo su
>> cd /opt/ttn-gateway/packet_forwarder
>> nano global_conf.toml
```

- Modifier l'adresse IP du Gateway-Bridge et enregistrez le fichier. Dans notre cas l'adresse IP est "localhost"

2.2 Chirpstack Gateway Bridge

Pour installer Chirpstack Gateway Bridge, il faut taper successivement ces commandes :

- Installations du broker mqtt

```
>> sudo apt install mosquitto
```

- Activer le repository de ChirpStack (i.e l'ajout des packages de ChirpStack à la base de l'OS).

```
>> sudo apt-key adv --keyserver keyserver.ubuntu.com
--recv-keys 1CE2AFD36DBCCA00
>> sudo echo "deb https://artifacts.chirpstack.io/packages
/3.x/deb stable main" | sudo tee /etc/apt/sources.list.d
/chirpstack.list
```

- Mise à jour de la base de l'OS

```
>> sudo apt update
```

- Installation de Chirpstack Gateway Bridge

```
>> sudo apt install chirpstack-gateway-bridge
```

- Démarrez le Chirpstack Gateway Bridge

```
>> sudo systemctl start chirpstack-gateway-bridge
```

NB : Le paramètre **start** peut etre remplacer par **stop**, **restart** ou **status** pour respectivement arrêter, redemarrer ou voir l'état courant du Chirpstack Gateway Bridge.

- Pour verifier si le Chirpstack Gateway Bridge a bien démarré, tapez :

```
>> systemctl status chirpstack-gateway-bridge
```

- Pour voir si tout machent pour le Chirpstack Gateway Bridge, tapez :

```
>> journalctl -u chirpstack-gateway-bridge -f -n 50
```

- Lancez Chirpstack Gateway Bridge au demarrage

```
>> sudo systemctl enable chirpstack-gateway-bridge
```

NB : Si l'on ne souhaite pas lancer Chirpstack Gateway Bridge au demarrage, il suffit de remplacer le paramètre **enable** par **disable**.

2.3 Chirpstack Network Server

Les commandes suivantes permettent d'installer chirpstack-Network-Server :

- Installez la base de données PostgreSQL (d'une version supérieure à la version 9.5)

```
>> sudo apt install postgresql
```

- Installez la base de données Redis (version minimum 2.6.0)

```
>> sudo apt install redis-server
```

- ChirpStack Network Server doit avoir sa propre base de données. Pour en créer une, démarrez l'interface de contrôle de PostgreSQL en tant que l'utilisateur postgres :

```
>> sudo -u postgres psql
```

- Une fois l'interface démarré, tapez les commandes suivantes :
 - créez l'utilisateur `chirpstack_ns` ayant pour mot de passe `'dbpassword'`

```
>> create role chirpstack_ns with login password 'dbpassword';
```

– créez la base de données `chirpstack_ns`

```
>> create database chirpstack_ns with owner chirpstack_ns;
```

– quittez l'interface

```
>> \q
```

- Pour vérifier si l'utilisateur et la base de données sont bien configurés, essayez de vous connecter en tapant :

```
>> psql -h localhost -U chirpstack_ns -W chirpstack_ns
```

- Installer ChirpStack Network Server en tapant :

```
>> sudo apt install chirpstack-network-server
```

- Après l'installation, accédez au fichier de configuration en tapant :

```
>> sudo nano /etc/chirpstack-network-server/chirpstack-network-server.toml
```

- Modifiez les lignes suivantes telles qu'elles si nécessaire.

```
dsn="postgres://chirpstack_ns:dbpassword@localhost/chirpstack_ns?sslmode=disable"
```



```
automigrate=true
```

```
name="EU_863_870"
```

```
timezone="Local"
```

```
server="tcp://127.0.0.1:1883"
```

- Enregistrez les modifications et quittez
- Démarrez Chirpstack Network Server

```
>> sudo systemctl start chirpstack-network-server
```

NB : Le paramètre **start** peut être remplacé par **stop**, **restart** ou **status** pour respectivement arrêter, redémarrer ou voir l'état courant du Chirpstack Network Server.

- Pour vérifier si Chirpstack Network Server a bien démarré, tapez :

```
>> systemctl status chirpstack-network-server
```

- Pour voir si tout marche pour Chirpstack Network Server, tapez :

```
>> journalctl -u chirpstack-network-server -f -n 50
```

- Lancez Chirpstack Network Server au démarrage

```
>> sudo systemctl enable chirpstack-network-server
```

NB : Si l'on ne souhaite pas lancer Chirpstack Network Server au démarrage, il suffit de remplacer le paramètre **enable** par **disable**.

2.4 Chirpstack Application Server

L'installation de Chirpstack Application Server se fait avec les commandes suivantes :

- ChirpStack Application Server doit avoir sa propre base de données. Pour en créer une, démarrez l'interface de contrôle de PostgreSQL en tant que l'utilisateur postgres :

```
>> sudo -u postgres psql
```

- Une fois l'interface démarré, tapez les commandes suivantes :
 - créez l'utilisateur **chirpstack_as** ayant pour mot de passe **'dbpassword'**

```
>> create role chirpstack_as with login password 'dbpassword';
```

– créez la base de données **chirpstack_as**

```
>> create database chirpstack_as with owner chirpstack_as;
```

– activez les extensions `trigram` et `hstore`

```
>> \c chirpstack_as
>> create extension pg_trgm;
>> create extension hstore;
```

– quittez l'interface

```
>> \q
```

- Pour vérifier si l'utilisateur et la base de données sont bien configurés, essayez de vous connecter en tapant :

```
>> psql -h localhost -U chirpstack_as -W chirpstack_as
```

- Installer ChirpStack Application Server en tapant :

```
>> sudo apt install chirpstack-application-server
```

- Générez un JSON Web Token (jwt) en tapant :

```
openssl rand -base64 32
```

Sauvegardez le jwt généré quelque part (supposons que le jwt est `e3+eD7zcVFJF3EFpPnM1oMj02DqUZxt5wR4IfPBpbtA=`).

- Accédez au fichier de configuration en tapant :

```
>> sudo nano /etc/chirpstack-application-server/chirpstack-
application-server.toml
```

- Modifiez les lignes suivantes telles qu'elles si nécessaire (ne pas oublier le jwt).

```
dsn="postgres://chirpstack_as:dbpassword@localhost
/chirpstack_as?sslmode=disable"
```

```
application_server.external_api.jwt_secret="e3+eD7zcVFJF3EFp
PnM1oMj02DqUZxt5wR4IfPBpbtA="
```

```
server="tcp://localhost:1883"
```

```
public_host="localhost:8001"
```

- Enregistrez les modifications et quittez
- Démarrez Chirpstack Application Server

```
>> sudo systemctl start chirpstack-application-server
```

NB : Le paramètre `start` peut être remplacé par `stop`, `restart` ou

`status` pour respectivement arrêter, redémarrer ou voir l'état courant du Chirpstack Application Server.

- Pour vérifier si Chirpstack Application Server a bien démarré, tapez :

```
>> systemctl status chirpstack-application-server
```

- Pour voir si tout marche pour Chirpstack Application Server, tapez :

```
>> journalctl -u chirpstack-application-server -f -n 50
```

- Lancez Chirpstack Application Server au démarrage

```
>> sudo systemctl enable chirpstack-application-server
```

NB : Si l'on ne souhaite pas lancer Chirpstack Application Server au démarrage, il suffit de remplacer le paramètre `enable` par `disable`.

Conclusion

De tout ce qui précède, on peut conclure que pour mettre en place un LoRaWAN, il faut beaucoup de composants matériels et logiciels. Les éléments dépendent les uns des autres. Il faut respecter rigoureusement les instructions afin d'obtenir un bon résultat ; puisse qu'une petite erreur peut être fatale.

Références

- [1] <https://www.blog-to-store.com/la-technologie-lora-partie-1-definition/>
consulté le 06/07/2020
- [2] <https://www.chirpstack.io/gateway-bridge/>
consulté le 27/07/2020
- [3] <https://www.chirpstack.io/gateway-bridge/install/requirements/>
consulté le 27/07/2020
- [4] <https://www.chirpstack.io/gateway-bridge/install/debian/>
consulté le 27/07/2020
- [5] <https://www.chirpstack.io/network-server/>
consulté le 27/07/2020
- [6] <https://www.chirpstack.io/network-server/install/requirements/>
consulté le 27/07/2020
- [7] <https://www.chirpstack.io/network-server/install/debian/>
consulté le 27/07/2020
- [8] <https://www.chirpstack.io/application-server/>
consulté le 27/07/2020
- [9] <https://www.chirpstack.io/application-server/install/requirements/>
consulté le 27/07/2020
- [10] <https://www.chirpstack.io/application-server/install/debian/>
consulté le 27/07/2020
- [11] <https://etcher.download/>
visité le 05/06/2020
- [12] <https://www.raspberrypi.org>
visité le 05/06/2020
- [13] <https://www.inpact-hardware.com/article/270/100830-comment-installer-simplement-raspberry-pi-sous-raspbian-avec-ssh-et-wi-fi-sans-ecran>
consulté le 27/07/2020
- [14] https://www.mobilefish.com/developer/lorawan/lorawan_quickguide_loraserver_installation.html
dernièrement consulté le 27/07/2020
- [15] https://www.mobilefish.com/download/lora/lora_part28.pdf
- [16] https://www.mobilefish.com/download/lora/lora_part28.1.pdf
- [17] https://www.mobilefish.com/download/lora/lora_part31.pdf
- [18] Lora-net/packet_forwarder [Internet]. LoRa ; 2020 [cité 29 déc 2020]. Disponible sur : https://github.com/Lora-net/packet_forwarder
- [19] <https://github.com/RAKWireless/RAK831-LoRaGateway-RPi>
- [20] <https://github.com/robertlie/RAK831-LoRaGateway-RPi>
- [21] <https://www.youtube.com/watch?v=bea7g5isD0w>