

Configurer une **gateway LoRa locale** avec un **Raspberry Pi** est un excellent projet pour montrer concrètement comment fonctionne un réseau LoRa privé. Voici les étapes détaillées pour mettre en place une gateway LoRa avec un Raspberry Pi et un module LoRa (comme le **Raspberry Pi + Dragino LoRa Hat** ou un module **SX127x**).

Matériel nécessaire

1. **Raspberry Pi** (modèle 3 ou 4 recommandé).
 2. **Module LoRa** (ex : Dragino LoRa Hat, SX1276).
 3. **Antenne LoRa** (compatible avec le module).
 4. **Carte microSD** (16 Go minimum) avec Raspbian OS installé.
 5. **Alimentation** pour le Raspberry Pi.
 6. **Câbles et accessoires** (clavier, souris, écran ou accès SSH).
-

Étape 1 : Préparer le Raspberry Pi

1. **Installer Raspbian OS :**
 - Téléchargez Raspbian Lite ou Desktop depuis le site officiel : <https://www.raspberrypi.org/software/>.
 - Flashez l'image sur la carte microSD avec un outil comme **Balena Etcher**.
 2. **Configurer le Raspberry Pi :**
 - Insérez la carte microSD dans le Raspberry Pi.
 - Connectez le clavier, la souris et l'écran, ou configurez l'accès SSH.
 - Allumez le Raspberry Pi et terminez la configuration initiale (langue, réseau, etc.).
-

Étape 2 : Installer le module LoRa

1. **Connecter le module LoRa :**
 - Si vous utilisez un **Dragino LoRa Hat**, placez-le directement sur les GPIO du Raspberry Pi.
 - Si vous utilisez un module **SX1276**, connectez-le aux broches GPIO (recherchez le schéma de branchement du module).
 2. **Installer les dépendances :**
 - Ouvrez un terminal sur le Raspberry Pi et mettez à jour les paquets :

```
sudo apt update
sudo apt upgrade
```
 - Installez les dépendances nécessaires :

```
sudo apt install git wiringpi
```
-

Étape 3 : Installer le logiciel de la gateway

1. **Télécharger le logiciel :**
 - Clonez le dépôt **LoRa Gateway** depuis GitHub :

```
git clone https://github.com/Lora-net/lora_gateway.git
cd lora_gateway
```
2. **Compiler le logiciel :**

- Compile le code source :
`make`
 - 3. **Configurer la gateway :**
 - Modifiez le fichier de configuration pour adapter les paramètres (fréquence, puissance, etc.) :
`nano global_conf.json`
-

Étape 4 : Tester la gateway

1. **Démarrer la gateway :**
 - Lancez le programme de la gateway :
`sudo ./lora_pkt_fwd`
 2. **Vérifier la communication :**
 - Utilisez un **capteur LoRa** ou un autre module LoRa pour envoyer des données à la gateway.
 - Vérifiez que les données sont bien reçues dans le terminal du Raspberry Pi.
-

Étape 5 : Configurer un réseau local

1. **Installer un serveur MQTT :**
 - Pour centraliser les données, installez un serveur MQTT comme **Mosquitto** :
`sudo apt install mosquitto mosquitto-clients`
 2. **Relier la gateway à MQTT :**
 - Modifiez le fichier de configuration de la gateway pour envoyer les données à un broker MQTT local.
 - Exemple de configuration dans `global_conf.json` :

```
"mqtt": {  
    "enabled": true,  
    "host": "localhost",  
    "port": 1883,  
    "topic": "lora/data"  
}
```
-

Étape 6 : Visualiser les données

1. **Utiliser Node-RED (optionnel) :**
 - Installez Node-RED pour visualiser et traiter les données LoRa :
`bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)`
 - Configurez un flux Node-RED pour recevoir les données MQTT et les afficher dans un tableau de bord.
 2. **Autres outils :**
 - Vous pouvez aussi utiliser des outils comme **Grafana** ou **InfluxDB** pour visualiser les données.
-

Étape 7 : Sécuriser la gateway

1. **Changer les mots de passe par défaut :**
 - Changez les mots de passe du Raspberry Pi et des services installés.
 2. **Configurer un pare-feu :**
 - Installez et configurez **ufw** pour limiter les accès :

```
sudo apt install ufw
sudo ufw allow ssh
sudo ufw enable
```
-

Résultat final

- Vous avez maintenant une **gateway LoRa locale** fonctionnelle qui peut recevoir des données de capteurs LoRa et les transmettre à un serveur MQTT ou une application locale.
 - Cela vous permet de créer un **réseau IoT privé** pour des applications comme la domotique, l'agriculture ou la surveillance industrielle.
-

Conseils supplémentaires

- Pour aller plus loin, vous pouvez configurer une **gateway LoRaWAN** avec **The Things Network** (TTN) pour connecter votre réseau local à un cloud public.
- Pour des démonstrations en cours, vous pouvez utiliser des **capteurs LoRa prêts à l'emploi** (ex : capteurs de température ou d'humidité).