Documentation LoRaWAN

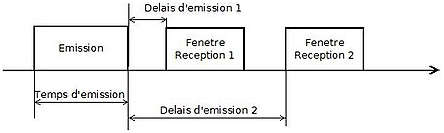
1. Le protocole LoRaWAN

Le protocole LoRaWAN a pour but de transmettre des données sur de longue distance, mais avec un débit faible limitant ainsi sa consommation en énergie et son prix car il est peu coûteux à implémenter grâce à son architecture en étoile et sa technique de modulation utilisant des fréquences libres (ne nécessitant pas de payer pour leur utilisation).

## **Les différentes classes de communication :**

La communication LoRa est bi directionnelle c'est-à-dire que la communication va dans les deux sens (équipement -> serveur et serveur –> équipement) et elle est définie en 3 classes:

* Classe A : Classe avec la plus basse consommation énergétique. Lors d’un envoi de données, il ouvre 2 fenêtres de réception successives durant lesquelles le serveur peut envoyer d’éventuels messages. Sinon, en dehors de ces fenêtres, le serveur ne peut rien envoyer à l’équipement.



*Figure 10 : fenêtre de réception de la classe A*

* Classe B : Consommation moyenne. Permet d’ouvrir des fenêtres de réception planifiées pour les messages provenant du serveur.

* Classe C : Classe avec la consommation la plus élevée. La fenêtre de réception est toujours ouverte.

La classe A doit être implémentée dans tous les équipements par souci de compatibilité. Les classes B et C sont des extensions.

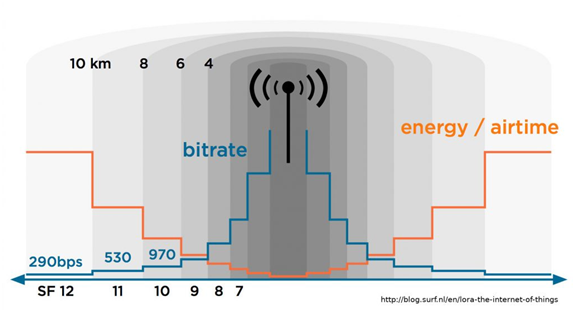
Pour le projet Air\_quality, nous avons utilisé la classe A car le cahier des charges précisé un système consommant le moins possible.

## 

## **Bande de fréquence :**

LoRaWAN utilise des fréquences dites libres. Ce qui signifie que n’importe qui peut les utiliser sans payer de droits de transmission. Le protocole LoRaWAN transmet via des fréquences radio basses avec une portée longue. Ces caractéristiques entraînent des contraintes spécifiques à chaque pays. En Europe, la bande de fréquence est entre 863 et 870 MHz mais, par défaut, on utilise généralement la fréquence 868MHz.

En fonction de la distance entre l’équipement et la passerelle, on peut régler un paramètre nommé facteur de propagation (SF : Spreading Factor) permettant de régler la vitesse de transmission.



*Figure 11 : Schéma des caractéristiques du facteur de propagation (SF)*

Sur la figure 11, on peut voir la portée et la vitesse de transmission pour chaque configuration du SF. Pour le projet Air\_quality, nous avons utilisé la configuration SF7, c'est-à-dire que la portée de communication est au minimum mais que la vitesse de transmission est au maximum.

## **Sécurité :**

Comme pour tout réseau, il faut garantir la confidentialité et la sécurité des données. Pour ce faire, le réseau LoRaWAN possède 2 clés de session réseau :

* Network Session key (NwkSKey) : Assure l’authenticité des équipements connectés au réseau
* Application Session Key (AppSKey) : Assure la sécurité et la confidentialité des données transmises

L’AppSKey est connu uniquement par l’utilisateur de l’application alors que la NwkSKey est fourni par l’opérateur réseau aux hébergeurs d’application autorisés (tel que le site TTN).

Pour obtenir ces deux clés, il existe deux méthodes :

* La méthode OTAA (Over-The-Air-Activation)
* La méthode APB (Activation By Personalization)

Pour que ces deux méthodes fonctionnent, Il faut 2 autres clés supplémentaires. La première s’appelle « Device EUI (DevEUI) », elle est configurée par le fournisseur de l’équipement et permet d’identifier l’équipement sur le réseau (vous pouvez l’obtenir grâce au projet Get\_devEUI:

<https://github.com/airqualitystation/firmware/tree/master/examples/Get_devEUI>).

La deuxième est l’ « Application EUI (AppEUI) » qui permet d’identifier l’application.

La différence entre ces deux méthodes est la réalisation et la sécurité car, pour la méthode APB, les clés de session NwkSKey et AppSKey ainsi que l’adresse de l’équipement, appelé « DevAddr », sont directement renseignés dans le programme de l’équipement LoRaWAN. De ce fait, il peut directement se connecter à l’application sur le serveur. Alors que pour la méthode OTAA, l’équipement doit effectuer une requête pour rejoindre le réseau à l’aide des clés utiles (DevEUI, AppEUI et AppKey). Les clés doivent être identiques sur l’application ainsi que dans le programme de l’équipement LoRaWAN (fichier Makefile du projet). Cette étape est importante car elle va permettre de rejoindre le réseau et de fournir à l’application et à l’équipement les 2 clés de sessions réseaux ainsi que la clé « Device Address » qui va être utilisée par l’équipement lors de la communication de données.

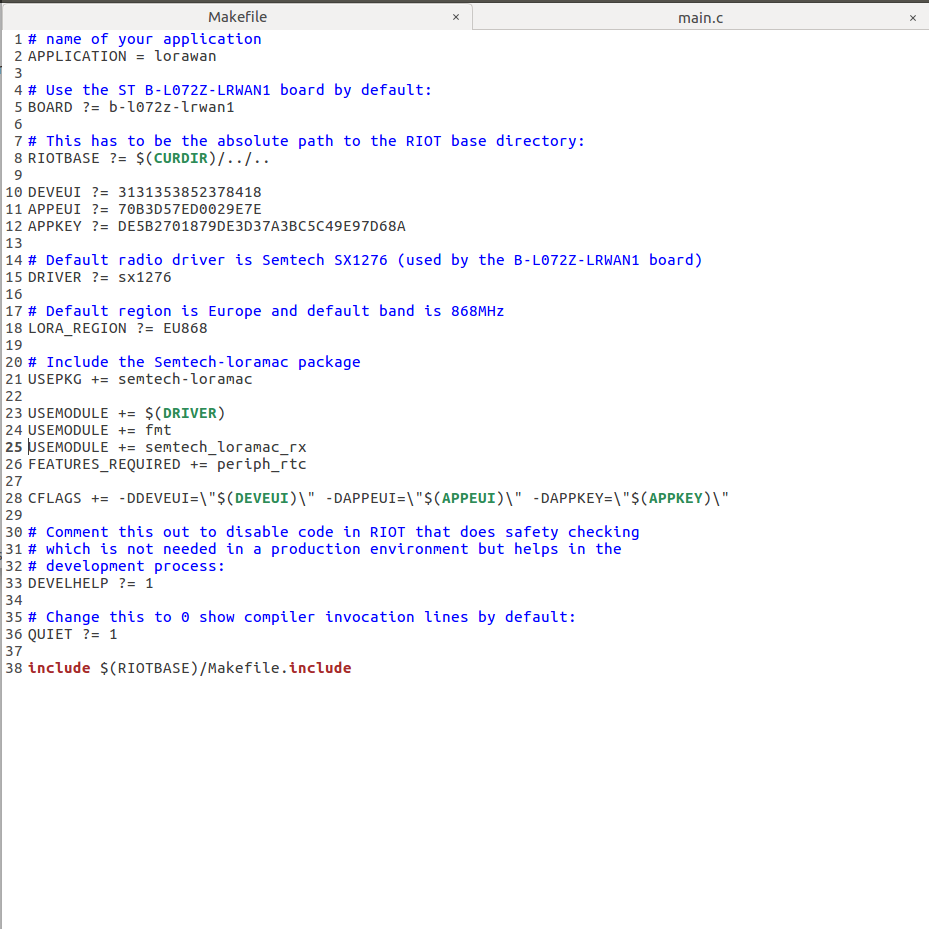
La méthode OTAA est donc plus compliquée à réaliser que la méthode APB. Cependant, elle est bien plus sécurisée car les clés de session sont différentes à chaque nouvelle connexion. Alors que pour l’autre méthode, les clés restent les mêmes.

1. RIOT OS

La première étape est d’installer RIOT OS. Une fois fait, on peut commencer à créer des projets.

Dans le dossier RIOT on peut trouver un dossier “examples” contenant un projet nommé lorawan. il est constitué d’un main.c et d’un fichier Makefile.

Le fichier Makefile contient tout la configuration LoRa et les bibliothèques utile au projet.



*figure 1 : Makefile du projet lorawan de RIOT OS*

On retrouve en premier le nom de l’application c’est à dire le nom dque va prendre le fichier exécutable. Ensuite il faut préciser la carte électronique qu’on utilise (ici la carte utilisé est la B-L072Z-LRWAN1). La liste des cartes disponible sur RIOT OS et disponible dans le dossier “boards”. On retrouve ensuite la configuration obligatoire pour la communication via le protocole LoRaWAN (pour connaître le DevEUI de sa carte vous pouvez utiliser le projet “Get\_devEUI” aussi dans le dossier examples, qui va vous retourner dans le terminal le numéro de votre carte. Attention il faut d’abord configurer le makefile avec la BOARD que vous utilisez !)

Pour l’appEUI et l’appKEY vous la trouverez sur le site qui récupère vos données envoyés, tel que TTN (The Things Network).

Pour la partie “driver”, notre carte utilise le SX1276 pour la radio fréquence.

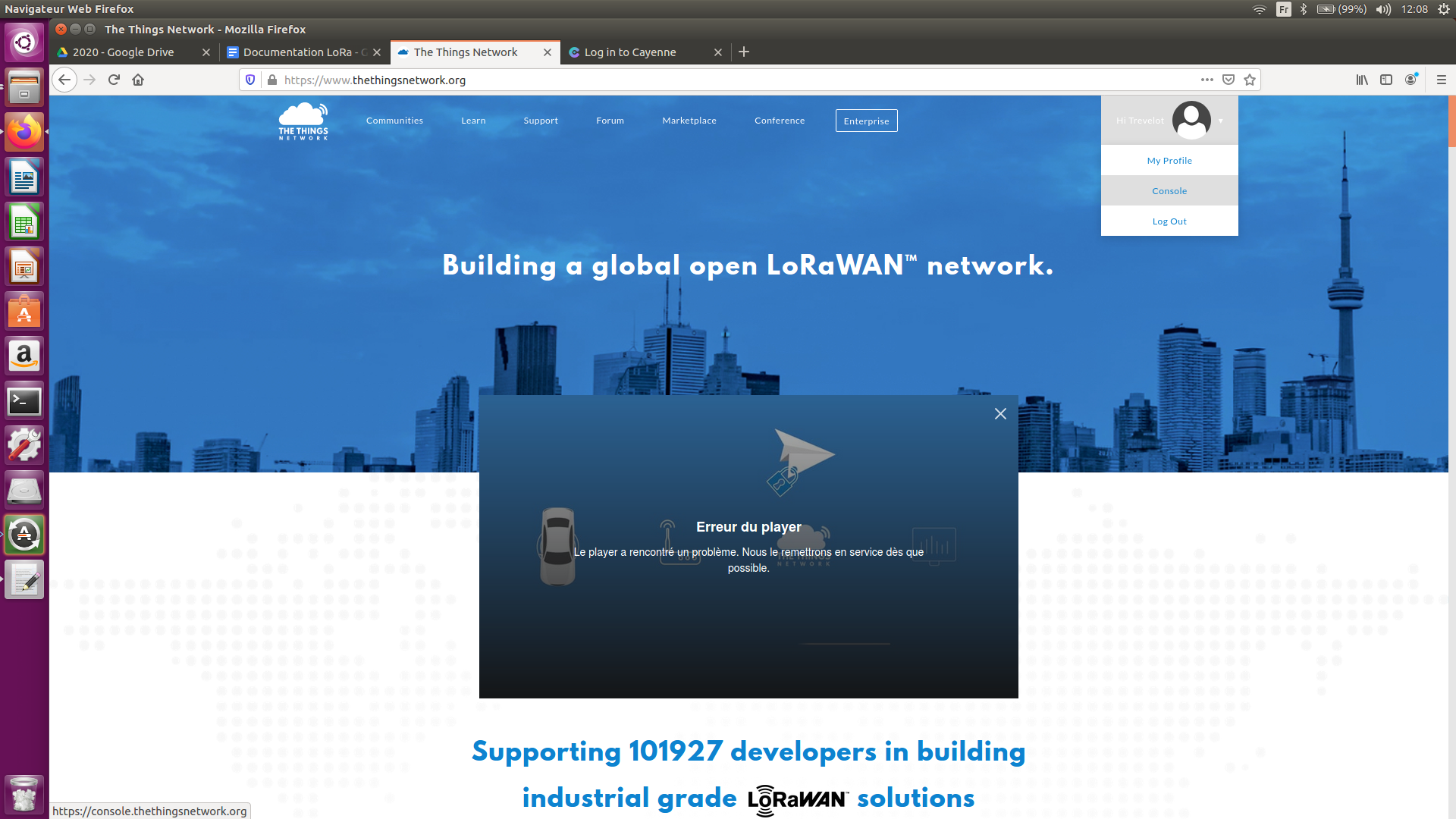
Les instructions commençant par “USE” signifie qu’on importe dans le projet des bibliothèques ou modules présent dans le dossier RIOT OS.

Ce projet permet de rejoindre une ou plusieurs gateway afin de transmettre des données via le protocole LoRaWAN. Il va créer un thread (processus) qui va tourner indéfiniment afin d’envoyer un message à une période défini dans le main.c

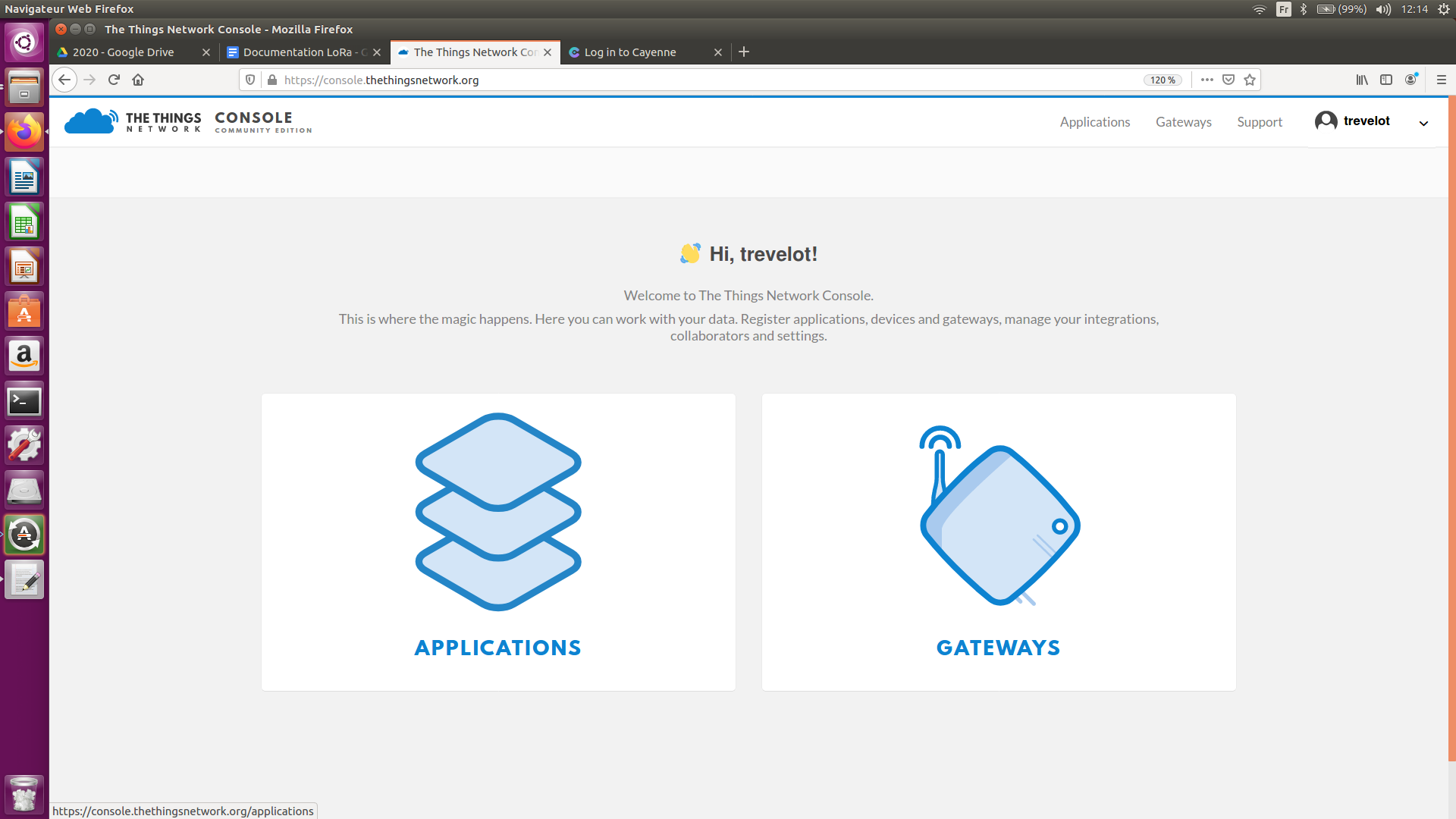
1. The Things Network

Pour la réception des données, nous utilisons le site TTN. La première étape est de se créer un compte et de se connecter.

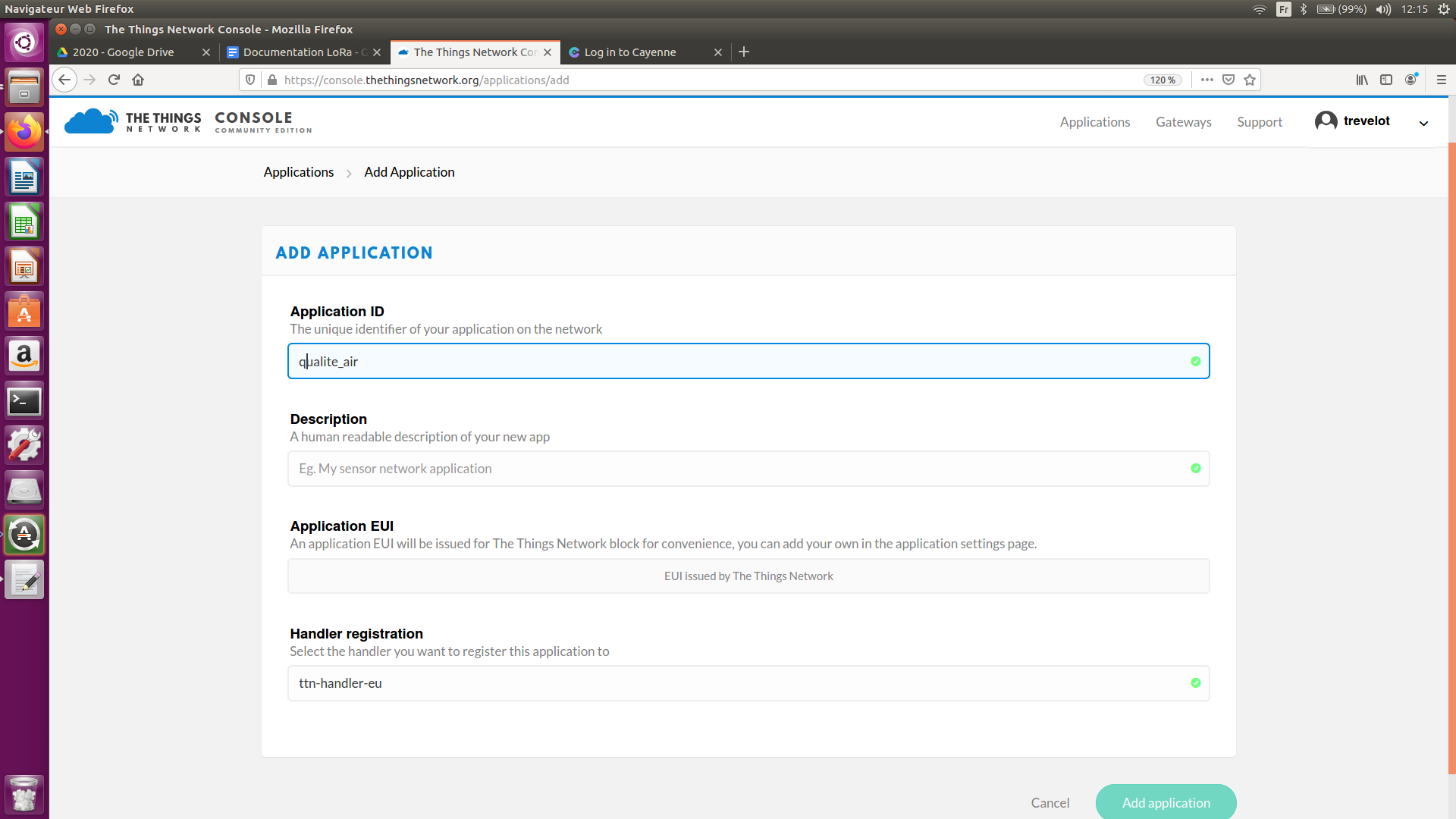
Ensuite il faut aller dans l’onglet console.



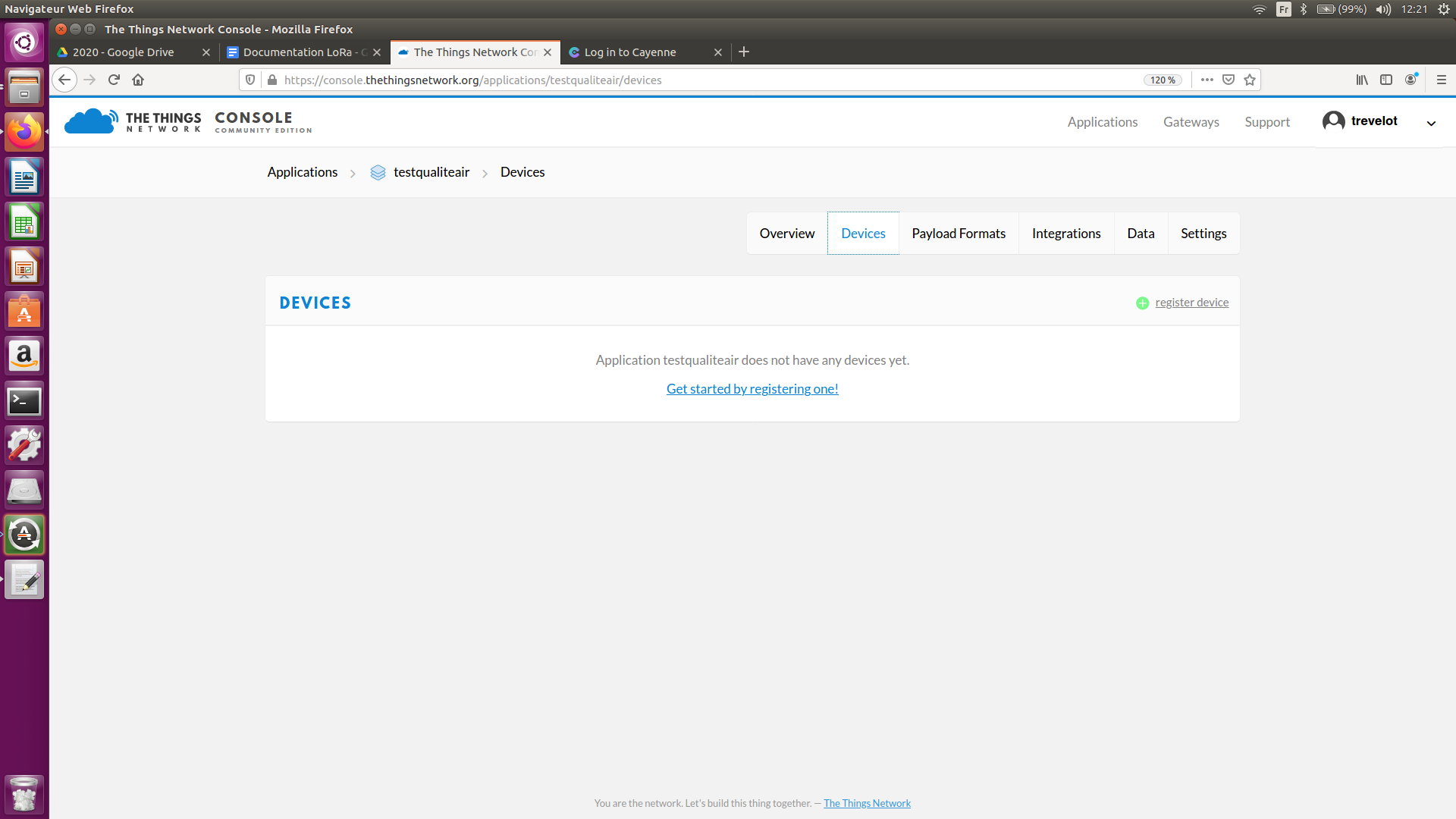
*1er étape : se connecter*



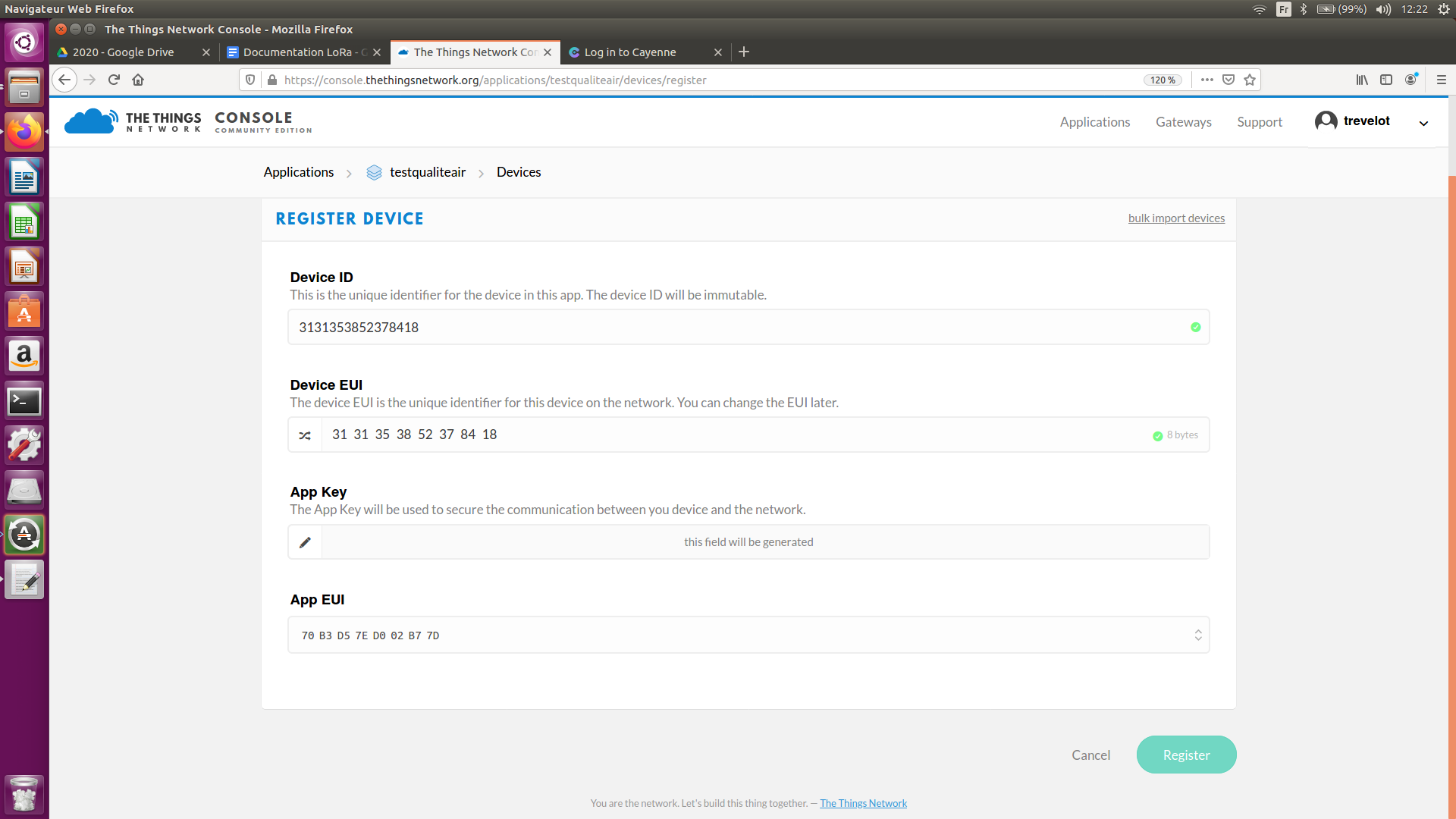
*2e étape : Cliquer sur Applications*



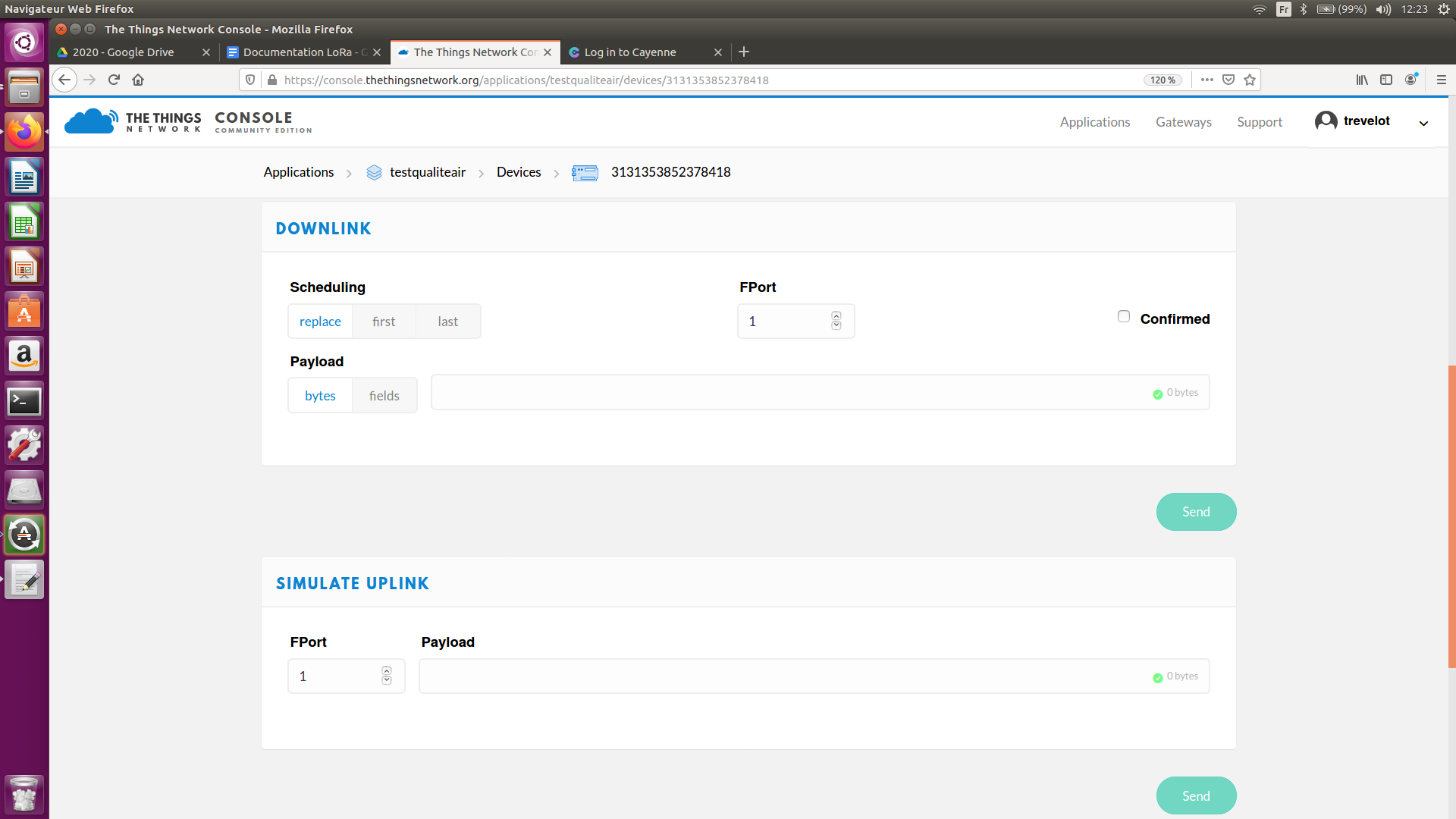
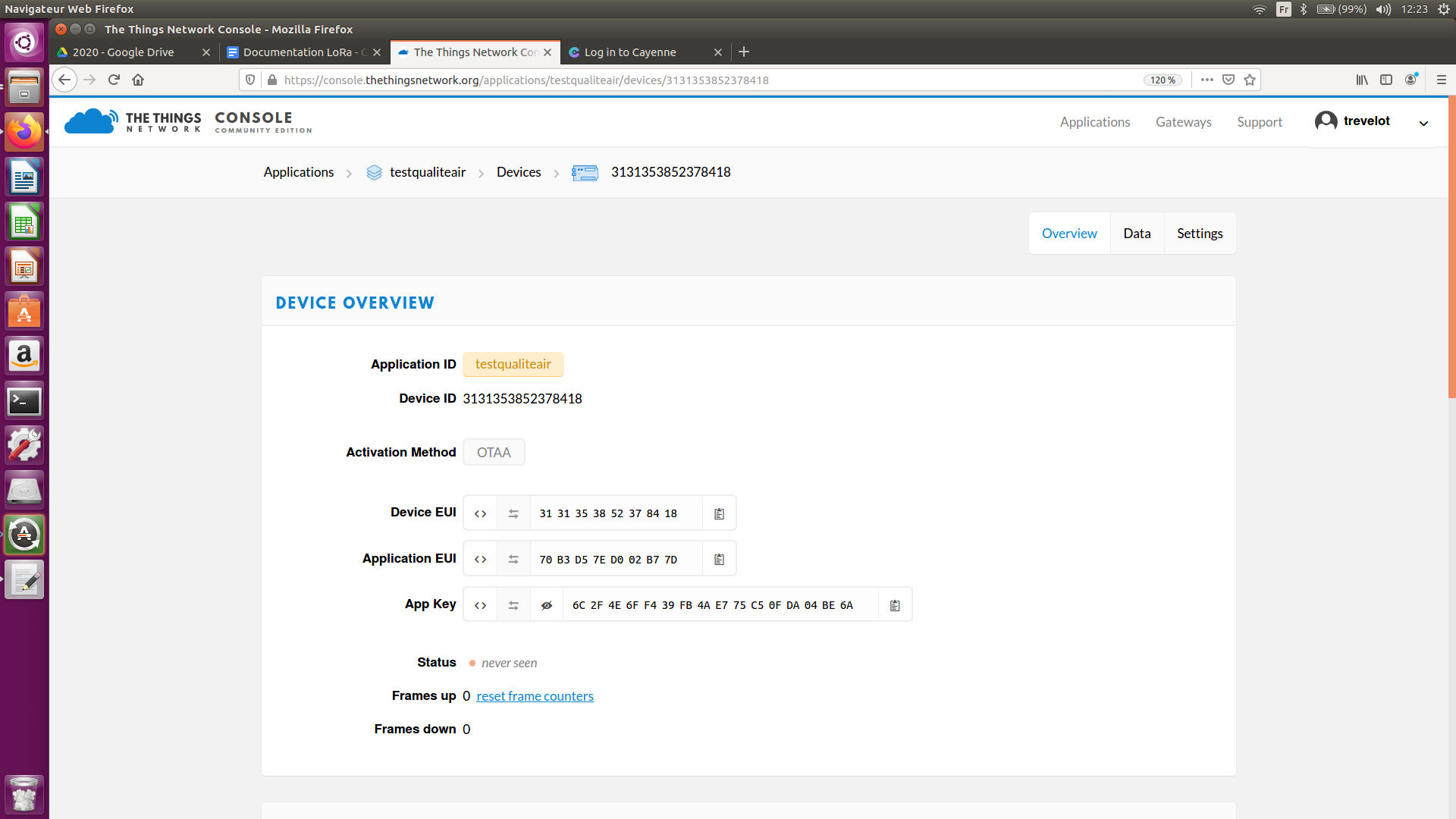
*3e étape : Créer une Application*



*4e étape : Enregistrer une carte électronique*



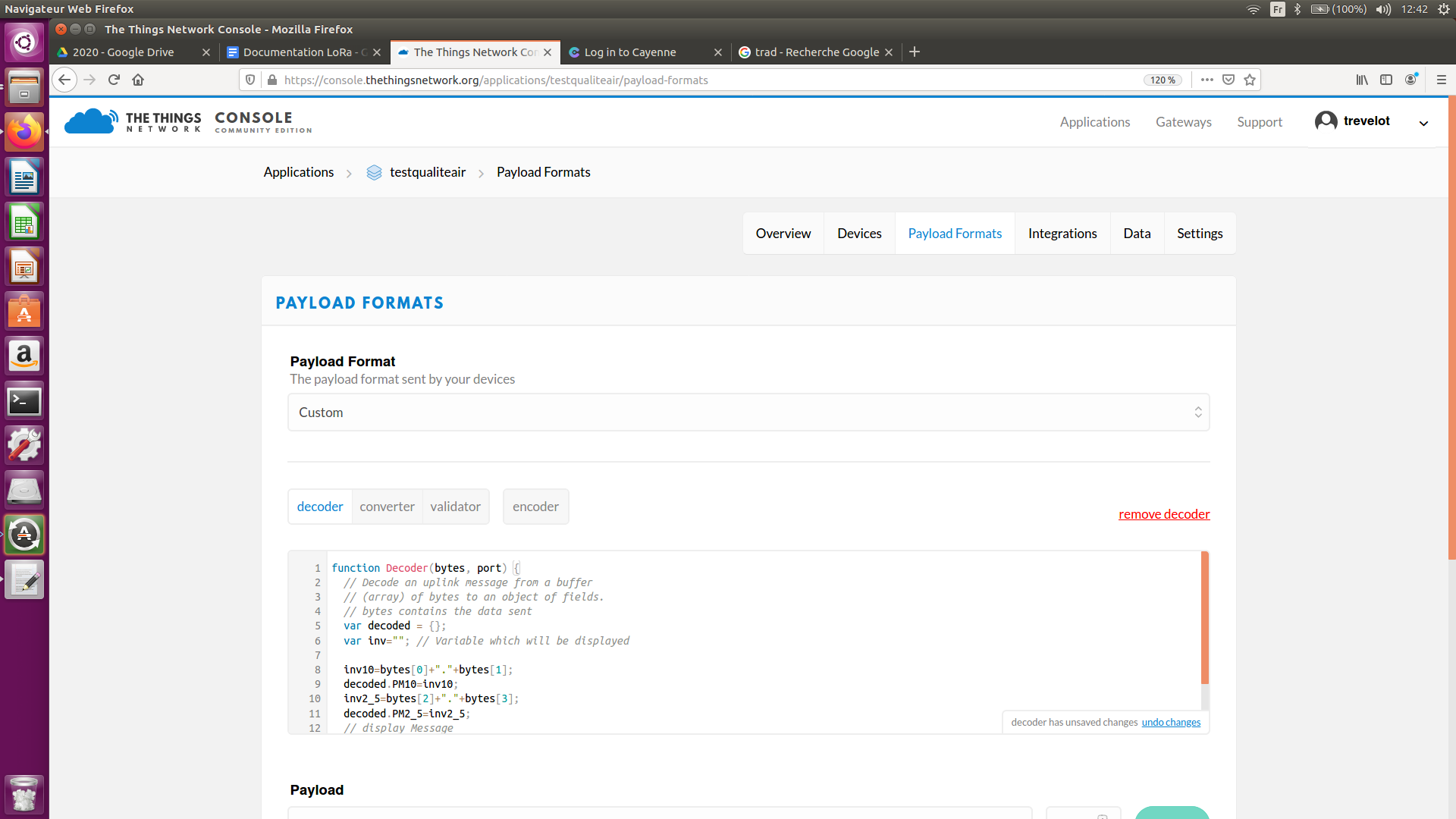
*5e étape : Entrer le numéro devEUI de la carte*



Dans l’onglet overview du Device on retrouve le type de sécurité choisi (OTAA est la plus sécurisé) ainsi que l’appEUI et l’appKEY. Un peu plus bas on retrouve le downlink qui permet d’envoyer un message à la carte.

Dans l’onglet Data, on peut voir toutes les données envoyés par la carte que nous venons de configurer à TTN.

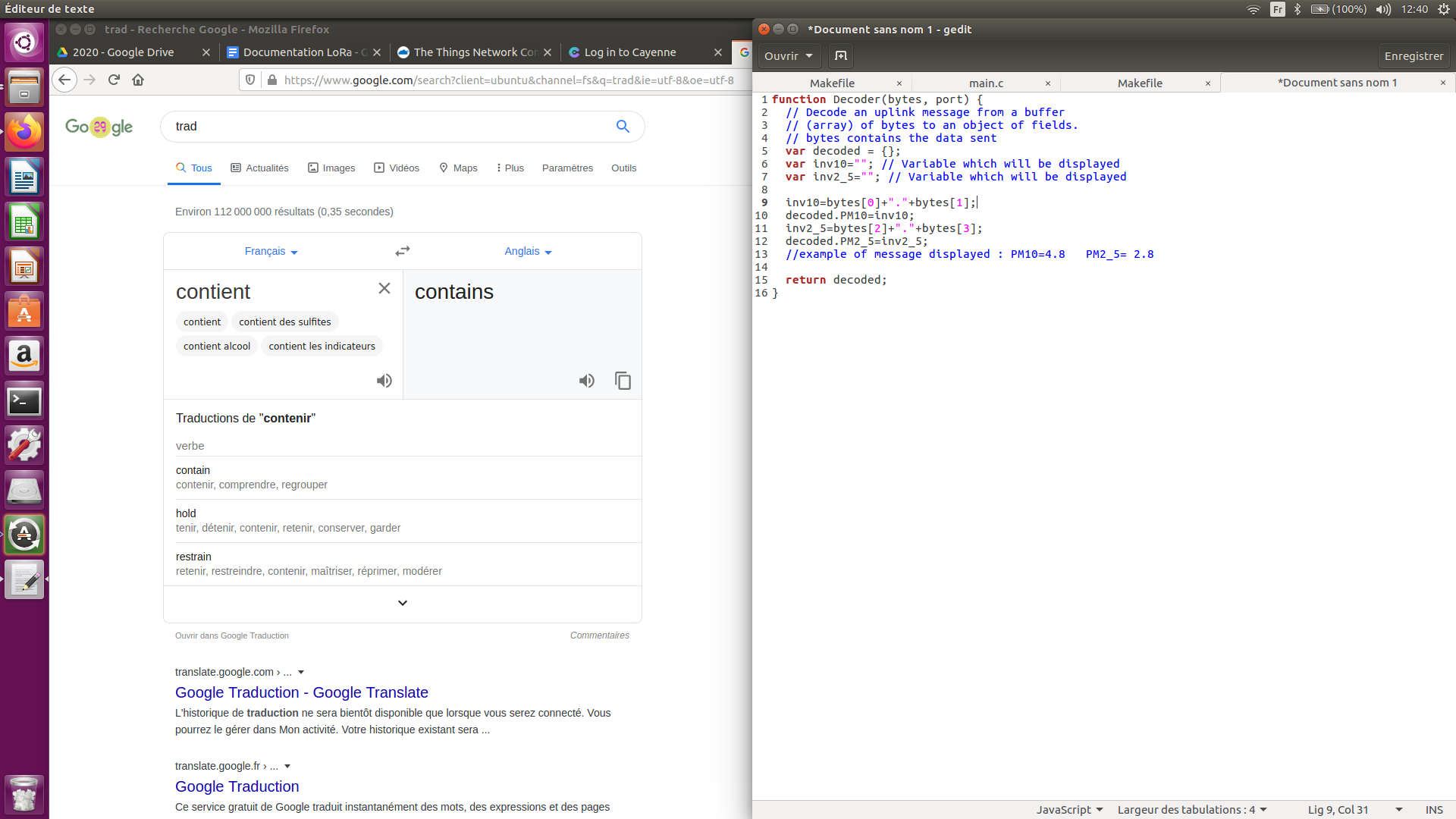
Avant d’envoyer des données il faut d’abord configurer la payload. Pour ca retourner dans votre application puis cliquer sur l’onglet “Payload Formats”.



*figure 2 : Onglet Payload Formats*

On retrouve l’onglet “decoder” qui va permettre de mettre en forme les données envoyés par la carte. Le programme est codé en Javascript.

Voici mon programme decoder :



*figure 3 : Fonction decoder de la Payload*

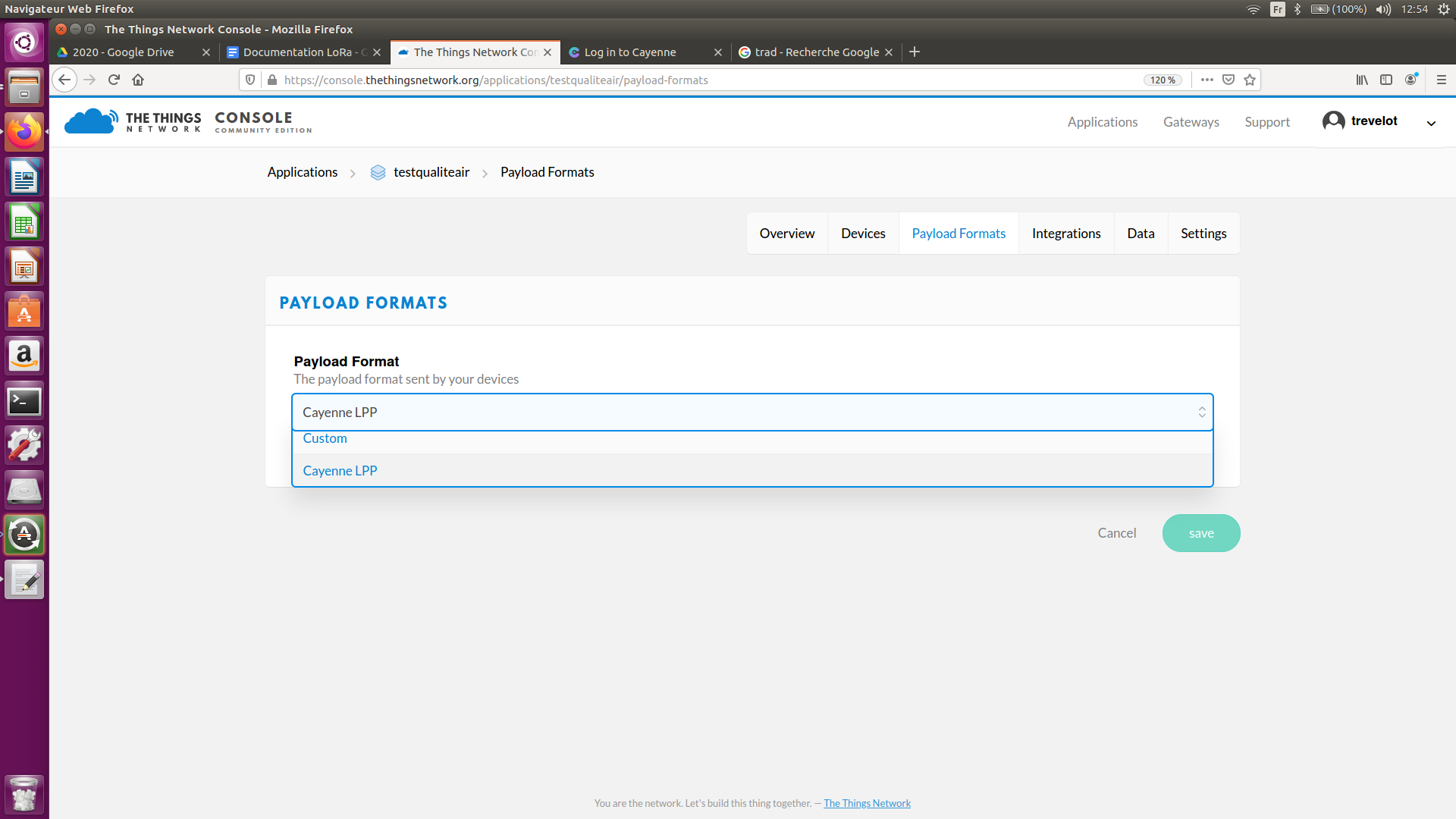
Le programme dépend de la trame envoyé. Ici j’envoi l’entier avant la virgule et l’entier après la virgule des données retournés par le capteur SDS011 (donc PM10 et PM2.5).

C’est pour ca que dans ma fonction, je remet en forme en plaçant une virgule ( +“.”+ ) entre les deux entiers.

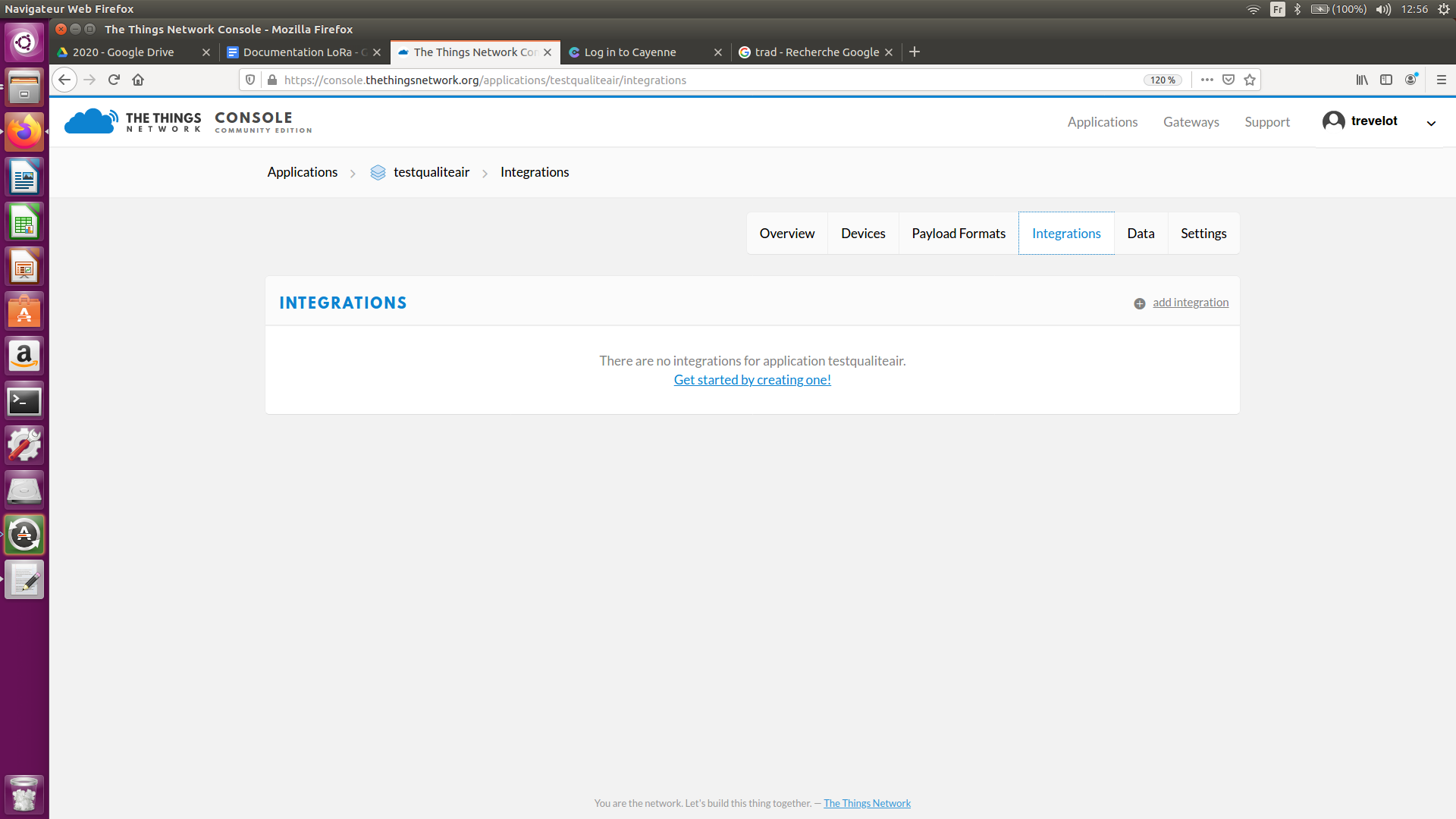
1. Cayenne LPP

Pour l’affichage des données, on peut aussi utiliser Cayenne lpp. Les avantages offerts par Cayenne sont sa facilité d’utilisation ainsi qu’un affichage clair et esthétique.

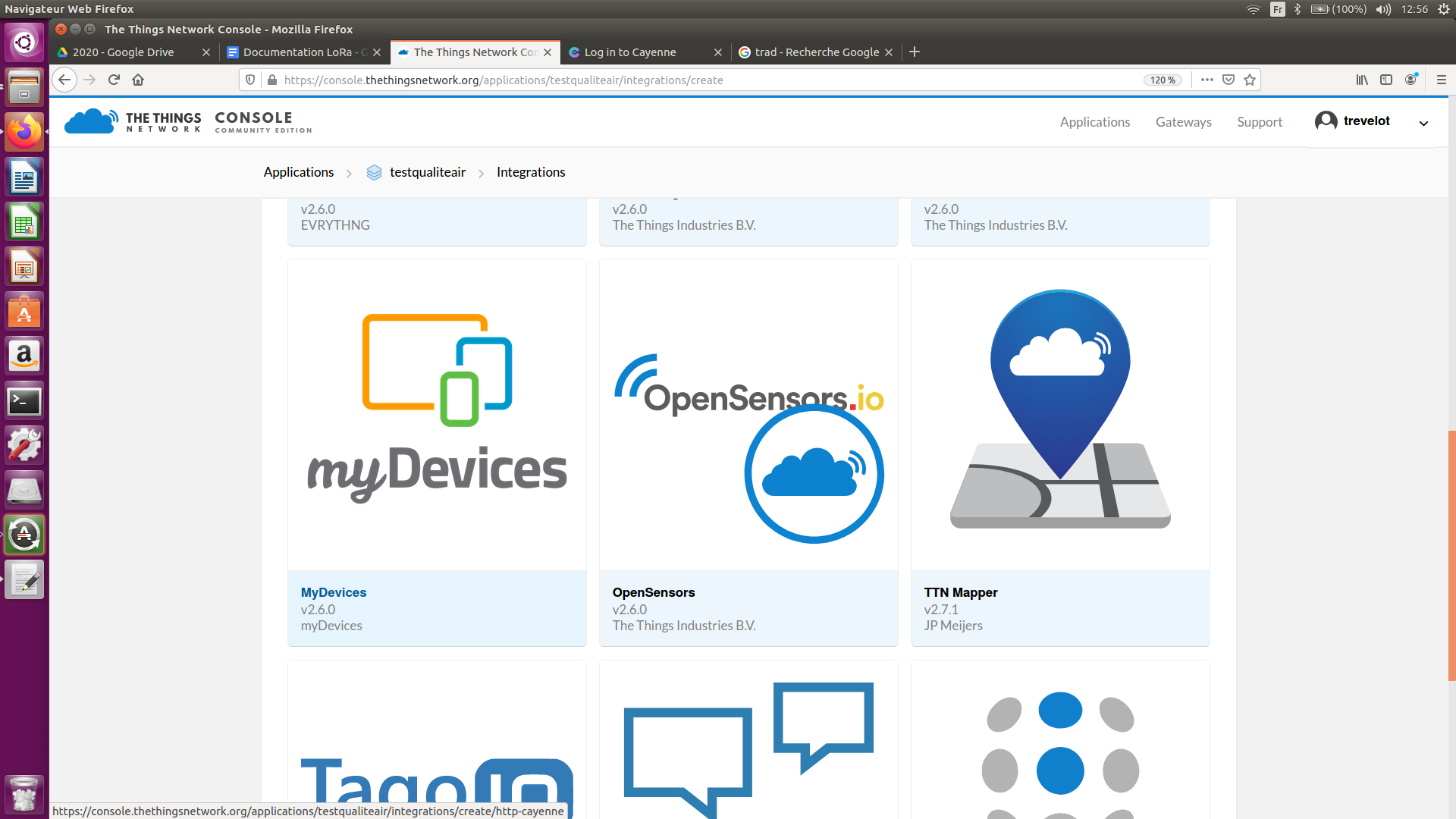
Pour l’obtenir, il faut mettre Cayenne LPP au lieu de custom dans l’onglet PayLoad Formats de votre application.



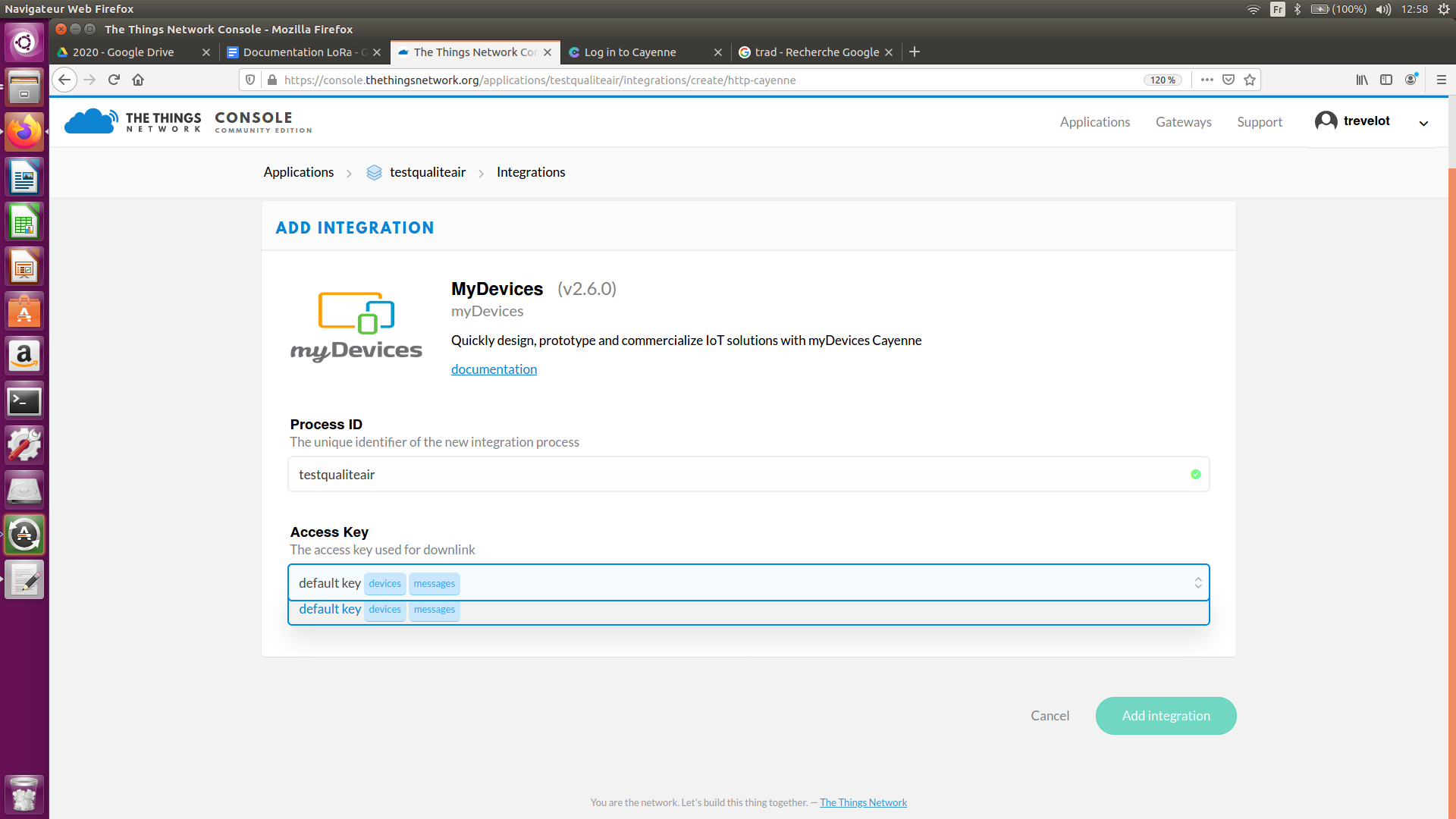
*1er étape : Indiquer Cayenne LPP dans Payload Formats*



*2e étape : Ajouter une Intégration*



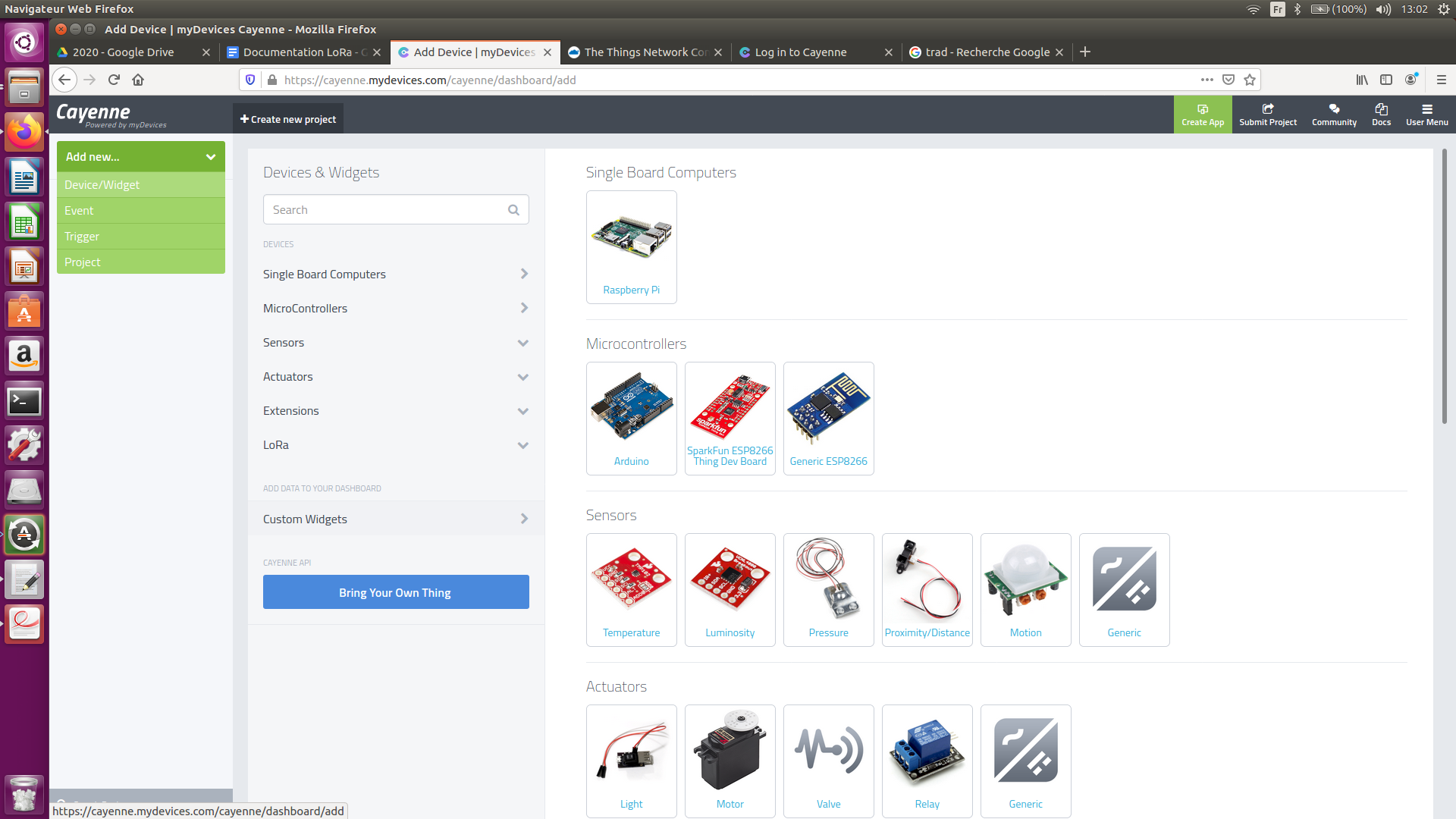
*3e étape : Cliquer sur MyDevices*



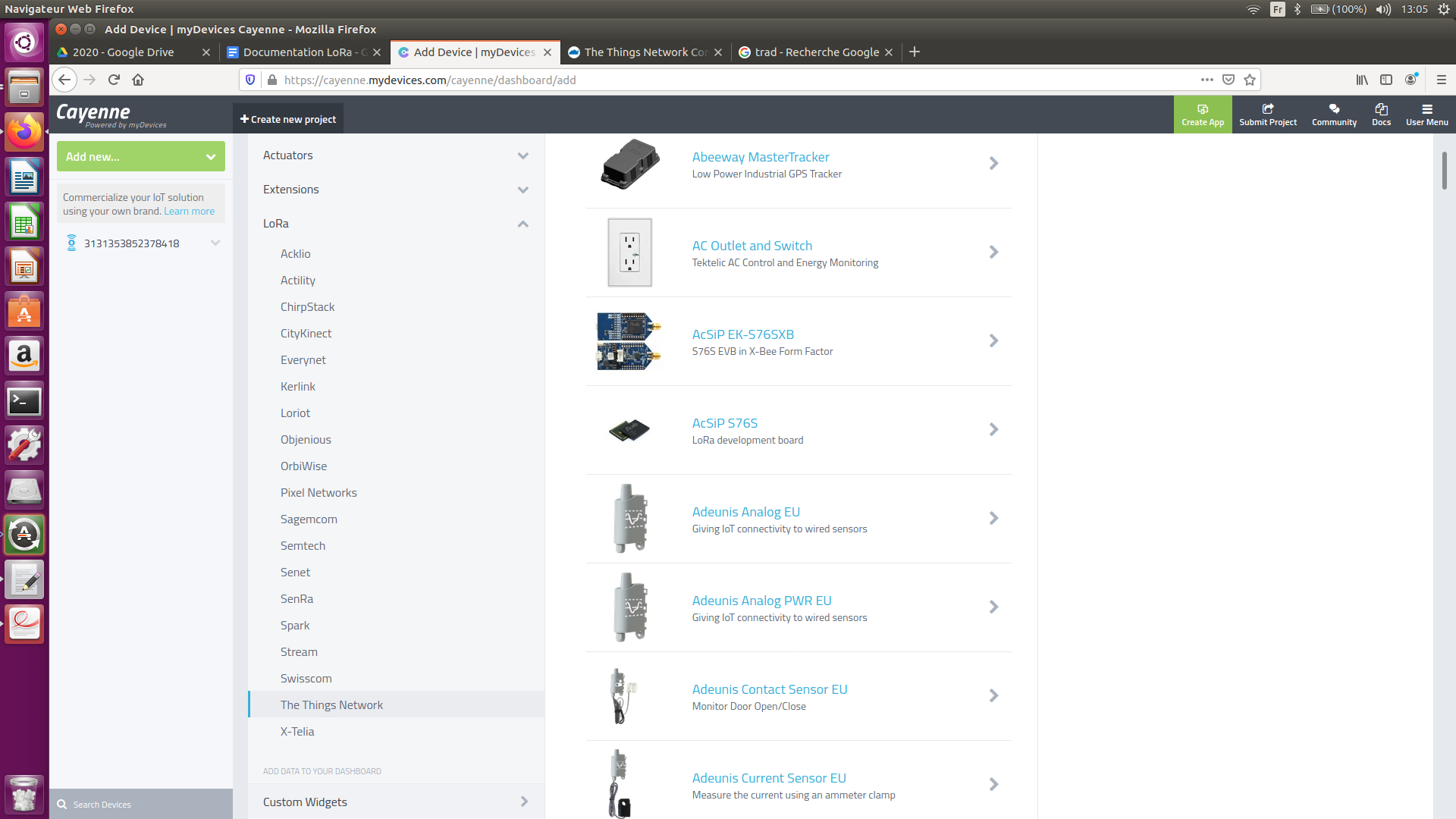
*4e étape : Donner un nom et sélectionner default key*

Pour la 5e étape, il faut se créer un compte sur le site de Cayenne puis se connecter :

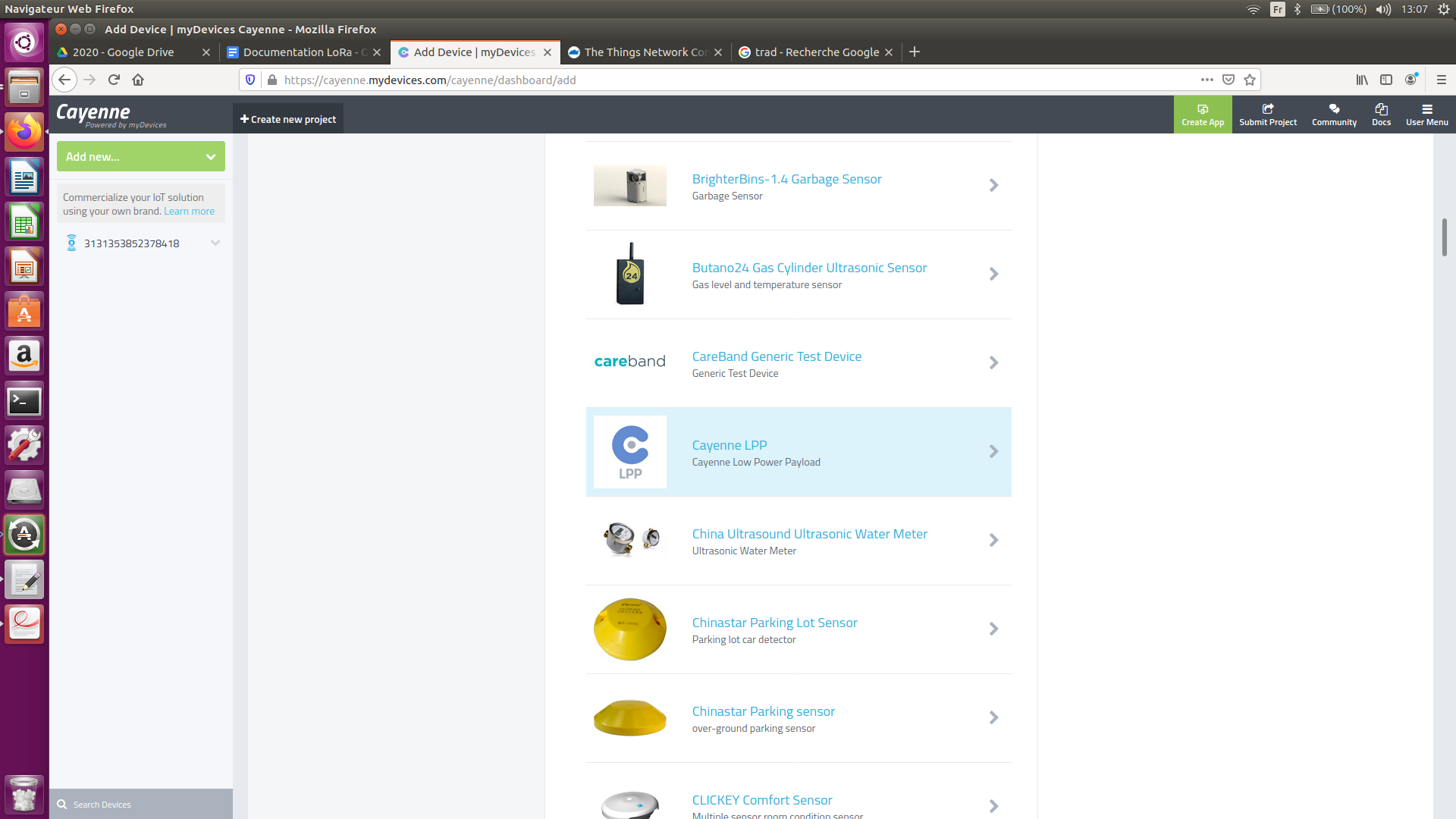
<https://cayenne.mydevices.com/cayenne/signup>



*6e étape : Ajouter un nouveau Device/Widget*



*7e étape : Cliquer sur LoRa puis sur The Things Network*

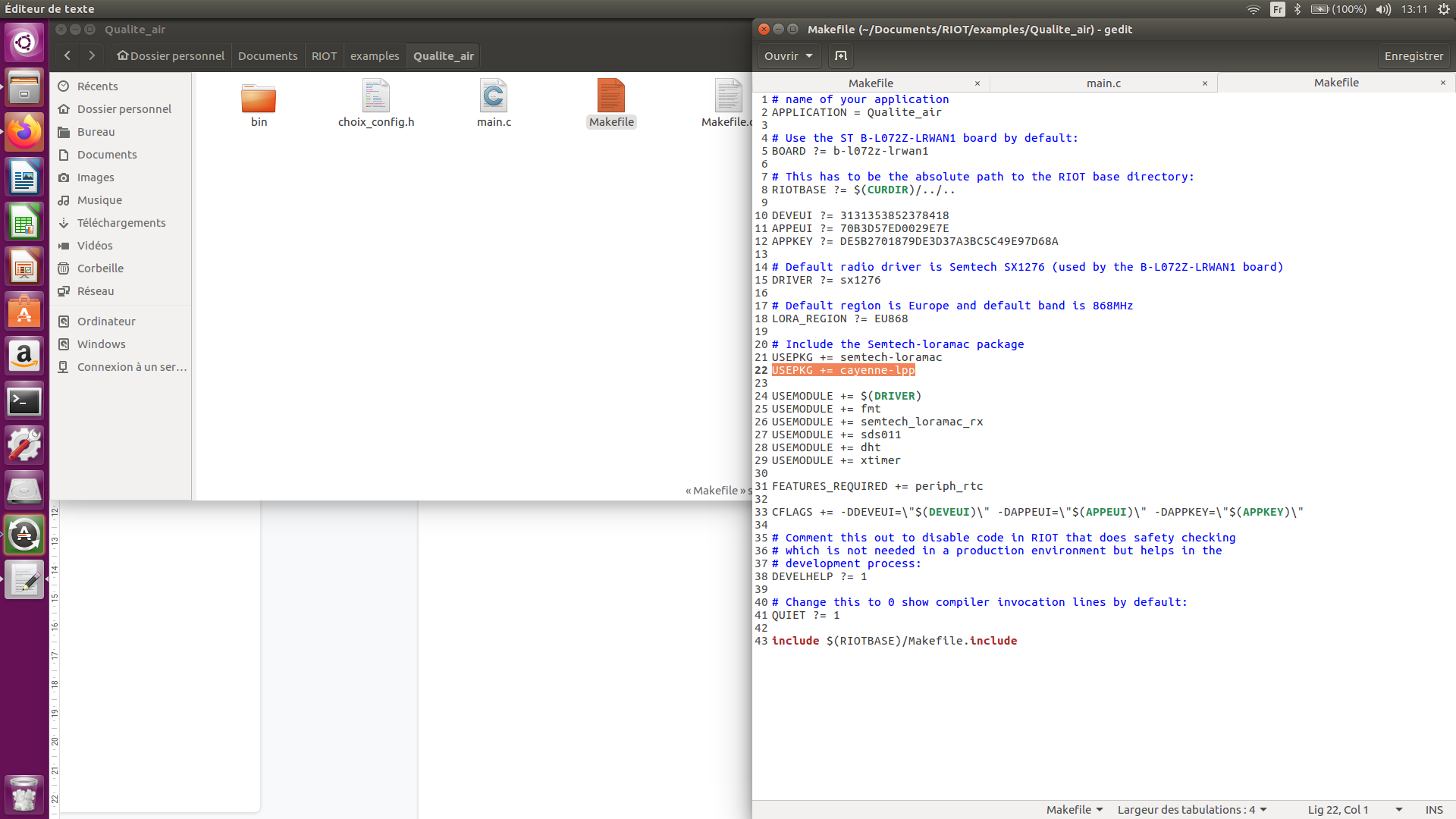


*8e étape : Sélectionner Cayenne LPP puis rentrer la devEUI de la carte*

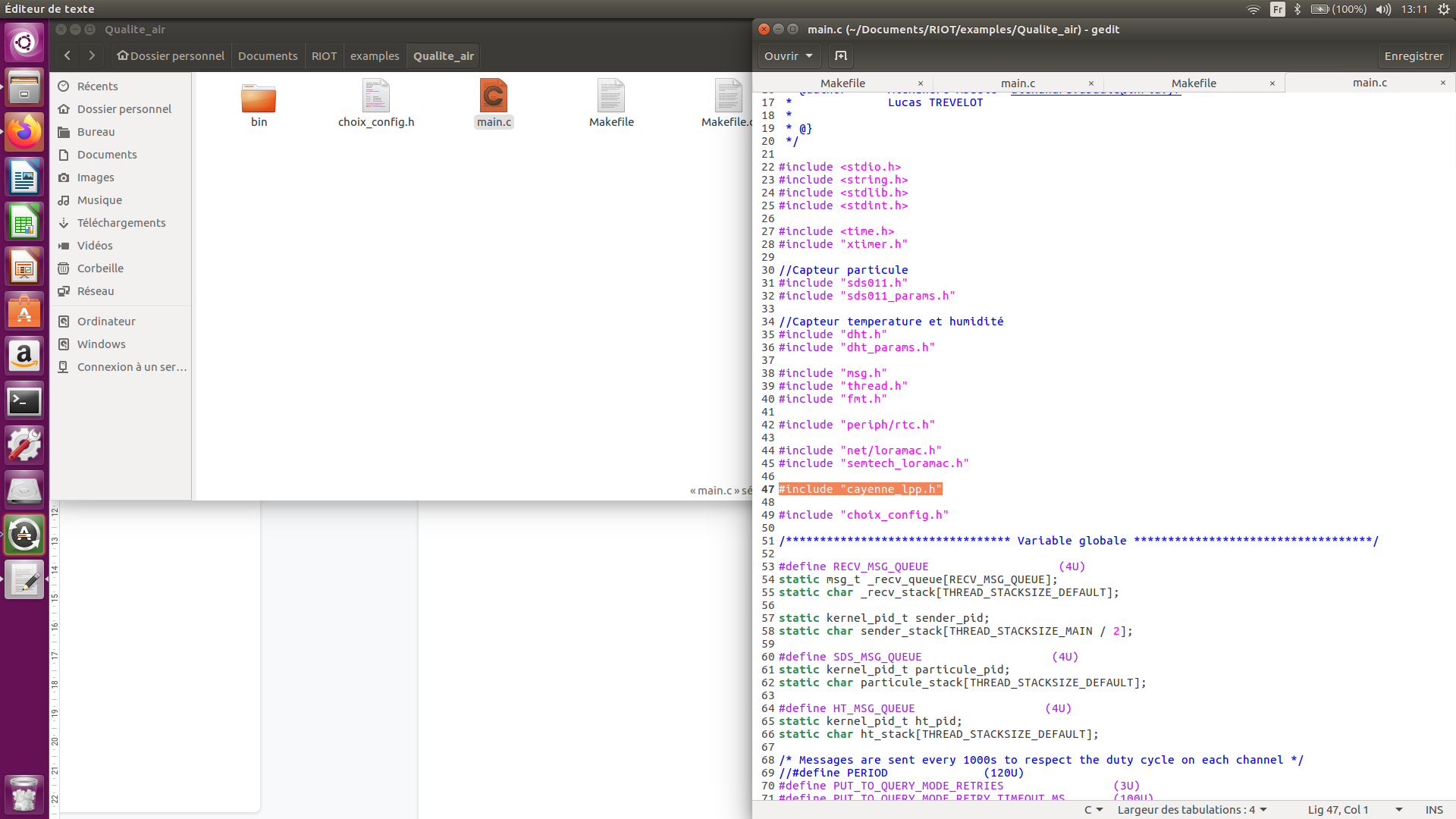
Maintenant que Cayenne est configuré, il faut ajouter la bibliothèque cayenne dans son projet RIOT. Pour ce faire il faut aller dans le fichier Makefile est ajouter la ligne :

USEPKG += cayenne-lpp

Puis dans le main.c, il faut ajouter : #include “cayenne\_lpp.h”

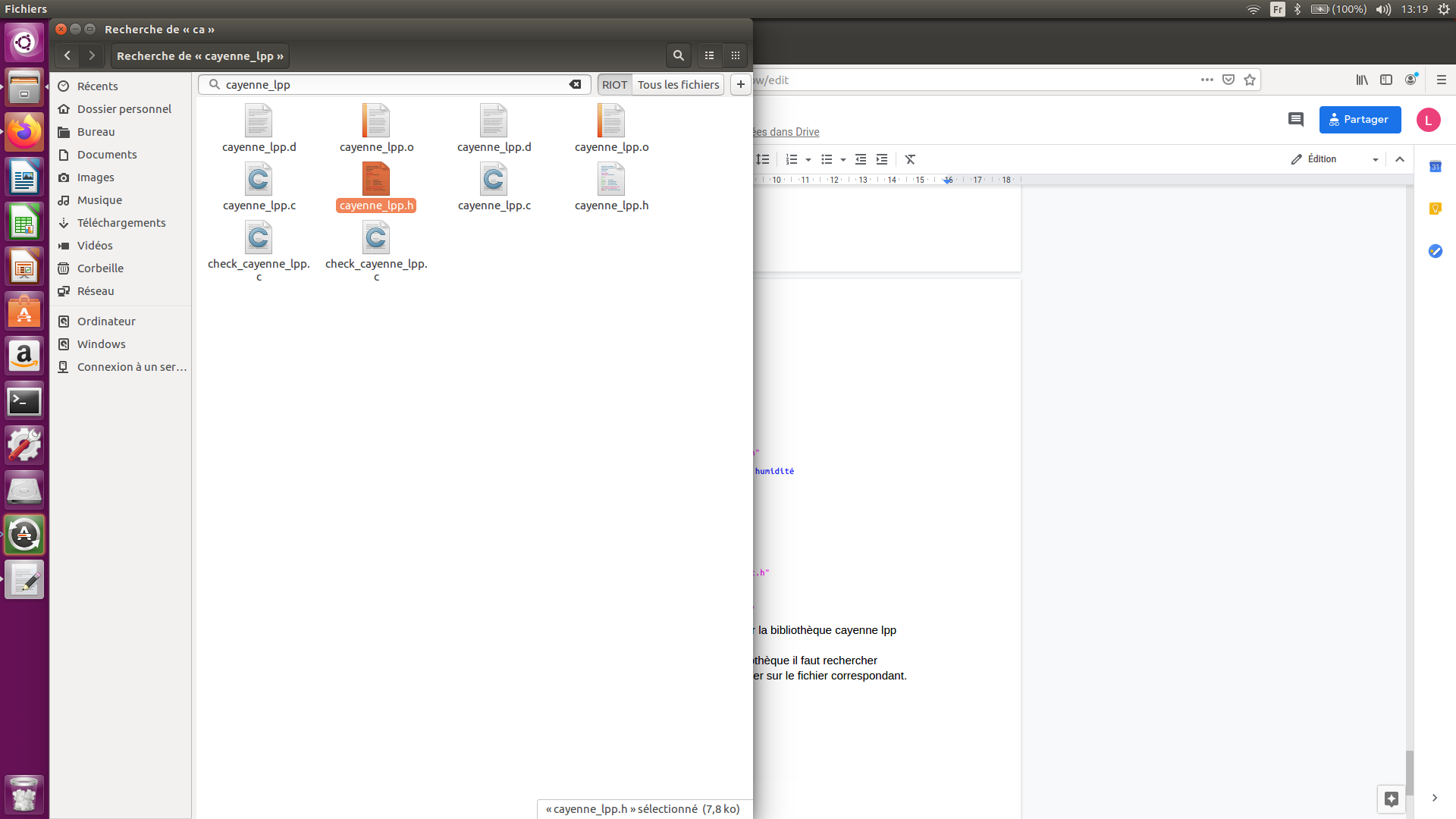


*figure 4 : Ligne dans le Makefile pour utiliser cayenne lpp*



*figure 5 : Ligne dans le main.c pour utiliser la bibliothèque cayenne lpp*

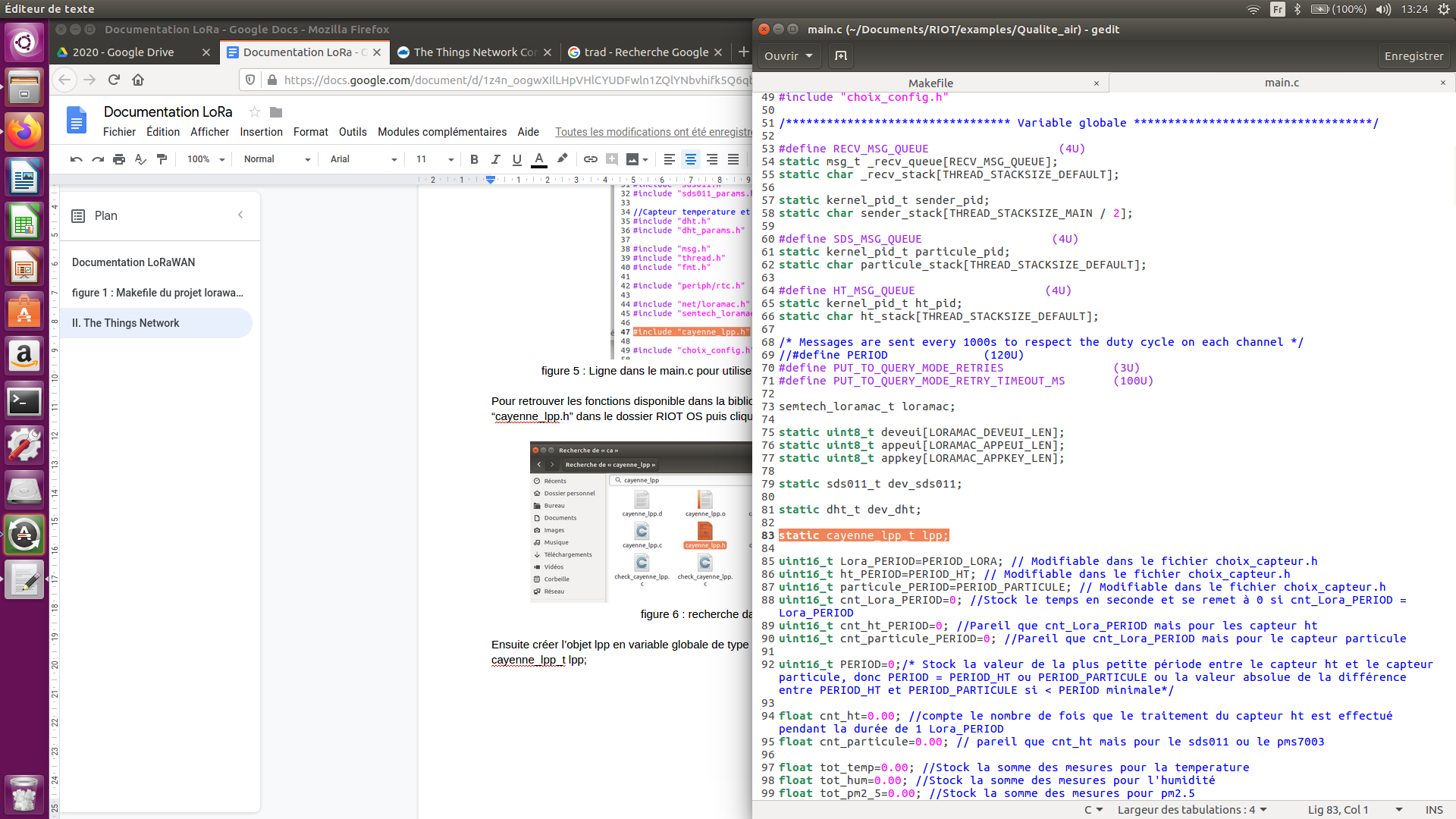
Pour retrouver les fonctions disponible dans la bibliothèque il faut rechercher “cayenne\_lpp.h” dans le dossier RIOT OS puis cliquer sur le fichier correspondant.



*figure 6 : recherche dans RIOT*

Ensuite, il faut créer l’objet lpp en variable globale de type cayenne\_lpp\_t :

*static cayenne\_lpp\_t lpp;*



*figure 7 : Variable globale lpp pour utiliser l’affichage sur cayenne*

Ajouter les valeurs des capteurs grâce au fonction de la bibliothèque.

exemple :

cayenne\_lpp\_add\_analog\_input(&lpp,1,valeur);

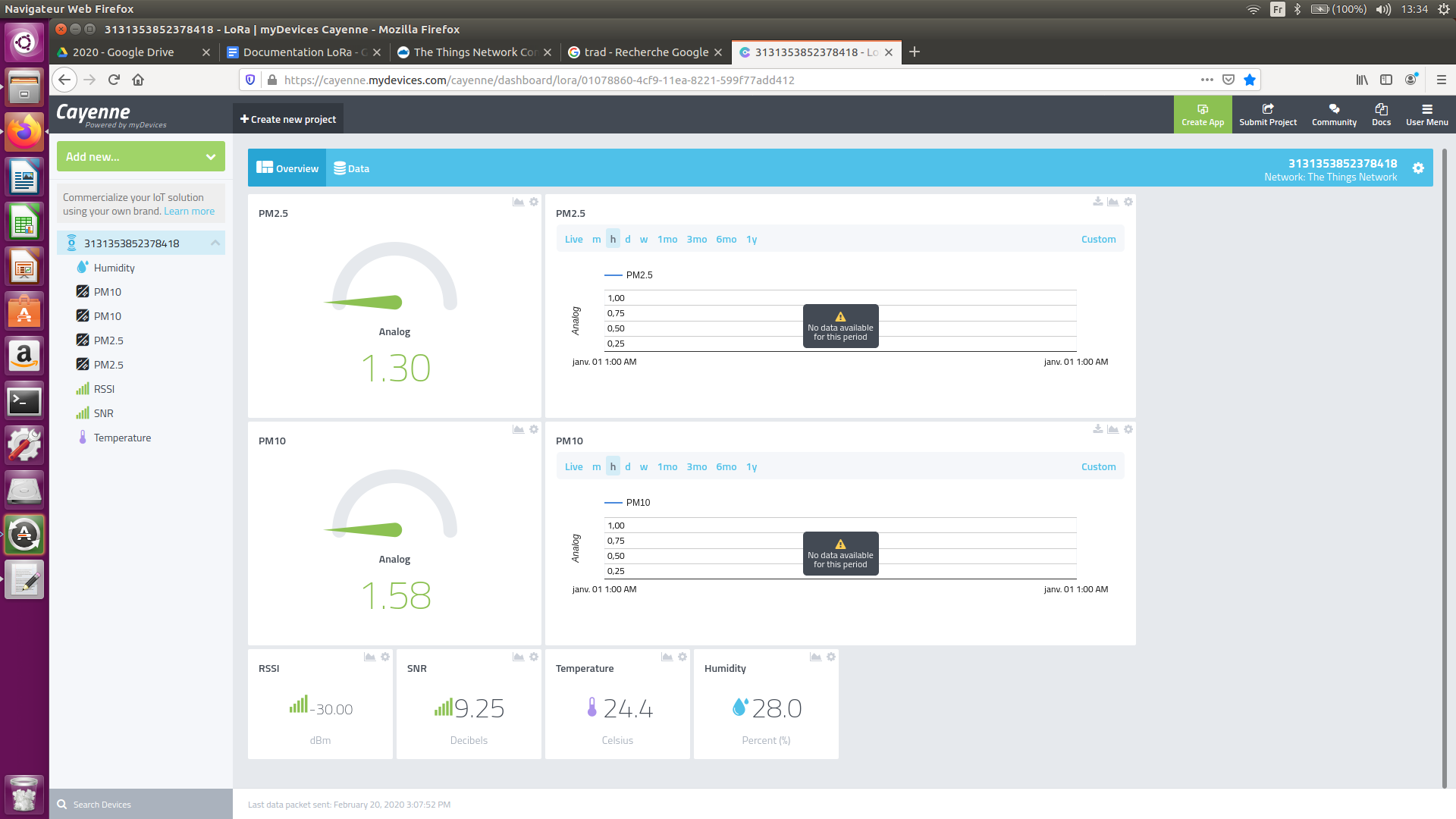
* objet de type cayenne\_lpp\_t : &lpp car ma variable s’appelle lpp
* valeur du port : ici pour affichage sur le port 1
* valeur que vous voulez afficher sur cayenne

Pour l’envoi via le protocole LoRaWAN :

semtech\_loramac\_send(&loramac,lpp.buffer,lpp.cursor);

* objet du type semtech\_loramac\_t
* tableau contenant les valeurs des capteurs
* taille du tableau

La dernière étape consiste à personnaliser votre interface sur le site de cayenne



*figure 8 : Affichage sur Cayenne*

1. Gateway



*figure 9 : The Things Indoor Gateway*

Une gateway permet donc de connecter un “device” à TTN via une connexion internet. Pour notre part, cette connexion internet est apporté par le wifi d’un de nos téléphones. C’est à dire que cette gateway devra être connecté à ce réseau internet pour pouvoir transmettre les messages de la carte vers TTN.

On ne peut pas privatiser une gateway dans le sens où n’importe quel device peut utiliser la gateway. C’est pour cela que certaines gateway peuvent être saturées.

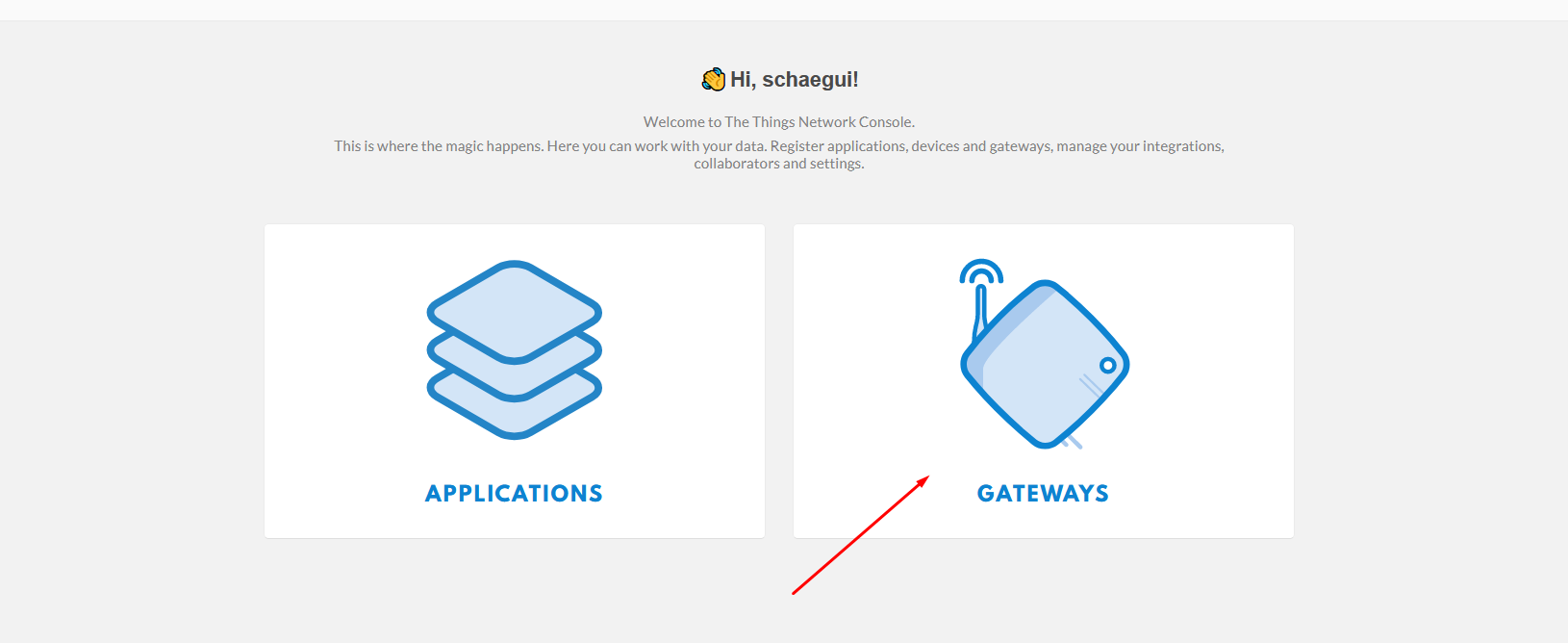
Cette gateway coûte environ 60 euros l’unité. Tout d’abord, il faut allumer la gateway et la relier à une connexion internet.

Pour cela, il faut suivre les étapes bien expliquées ici : <https://www.thethingsnetwork.org/docs/gateways/thethingsindoor/>

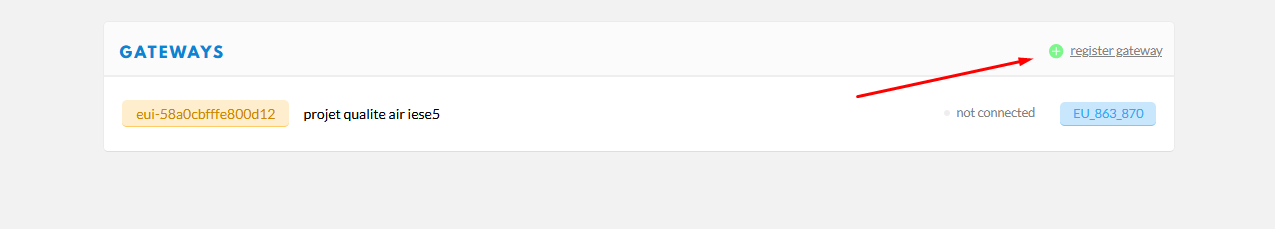
Ils expliquent comment activer la gateway puis comment la relier à internet et puis l’ID que l’on doit entrer dans TTN pour connecter la gateway au réseau TTN.

Nous avons utilisé la wifi d’un téléphone comme connexion internet pour la gateway mais on peut prendre n’importe quelle connection à internet. Sachant que le fonctionnement de la gateway dépend fortement de celle ci.

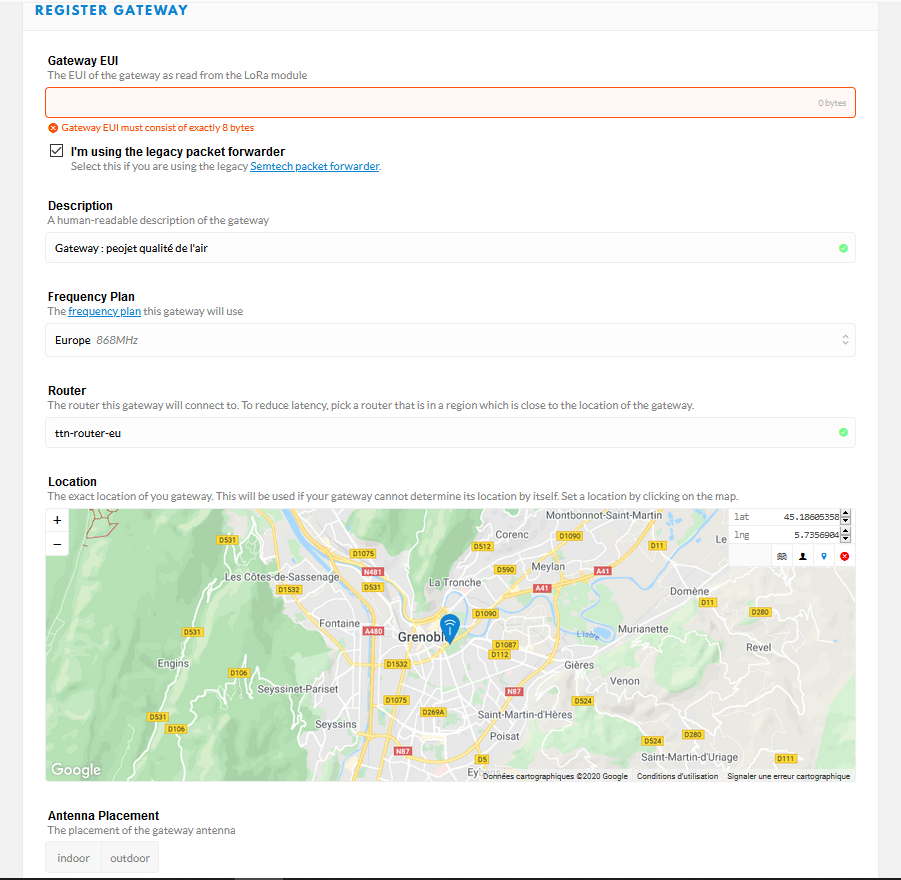
Tout comme un device, il faut avoir un compte TTN et créer une gateway. Pour cela, il faut aller sur la page TTN puis dans console. On arrive sur cette page. On clique sur gateways :



Puis sur register gateway :



On arrive ensuite sur une page ou on va entrer des informations sur notre gateway.



Le Gateway ID est l’ID que notre gateway a, cet ID se trouve directement sur la gateway.

Il faut cocher la case “I’m using the legacy packet forwarder”.

(revoir la section **Connection to the The Things Network Backend** se trouvant dans le lien ci dessus)

La description sert à décrire à quoi va servir votre gateway.

Bien spécifier la fréquence utilisée. En France, on utilise Europe 868 MHz ainsi que la localisation de votre gateway.

Voilà, normalement, votre gateway est connectée à internet et au réseau TTN et on peut voir que des cartes passent par cette gateway dans l’onglet “trafic”.