Tests IoT pour l'IuT

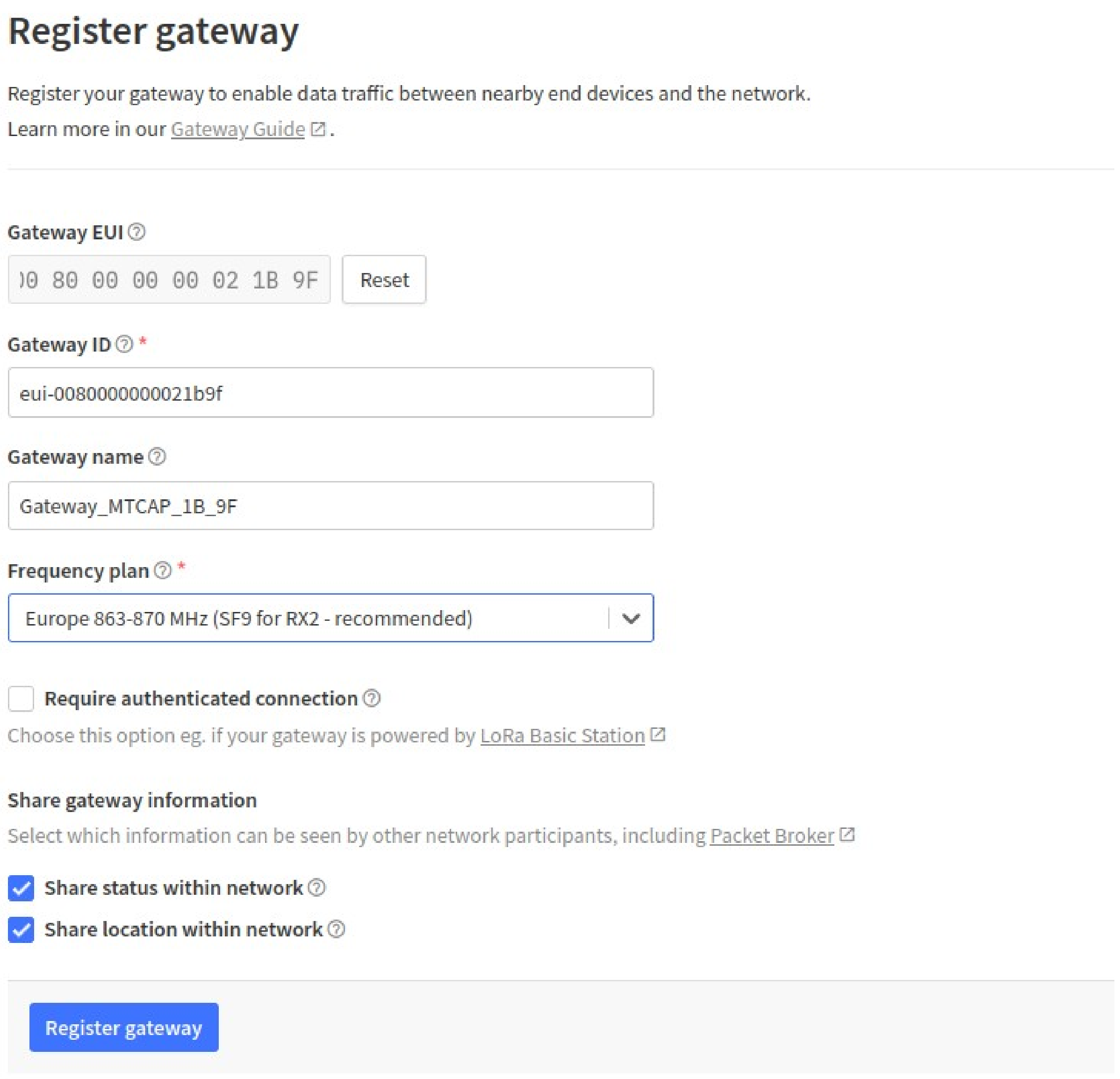
Table des matières

1. The things network

1.1. Create account

J'ai dû utiliser mon adresse gmail perso, mon adresse pro ne fonctionnait pas.

1.2. Register gateway



Register gateway

Register your gateway to enable data traffic between nearby end devices and the network.
Learn more in our Gateway Guide.

Gateway EUI
)0 80 00 00 00 02 1B 9F

Gateway ID *
eui-0080000000021b9f

Gateway name
Gateway_MTCAP_1B_9F

Frequency plan *
Europe 863-870 MHz (SF9 for RX2 - recommended)

□ Require authenticated connection

Choose this option eg. if your gateway is powered by LoRa Basic Station

Share gateway information

Select which information can be seen by other network participants, including Packet Broker

☑ Share status within network

☑ Share location within network

2. Configuration Gateway Multitech MTCAP-868-041A

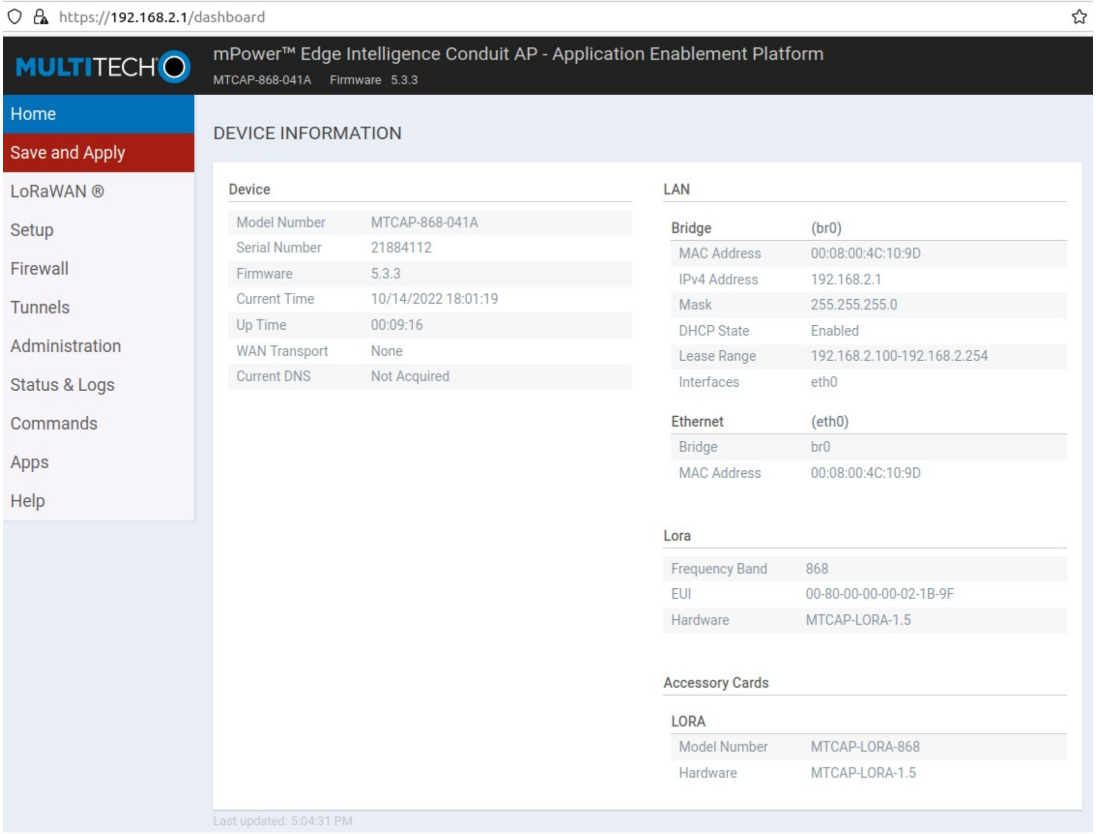
2.1. Première connexion

On met une ip fixe au PC et on se raccorde en IP fixe à l'interface en 192.168.2.1/24

On définit un user admin et un mdp :

```
username: iutbeziers
password : C15n3t@L15r1_BZRs
```

On arrive sur l'interface, le wizzard se lance. On fait next / next / next et on arrive ici :



mPower™ Edge Intelligence Conduit AP - Application Enablement Platform

MTCAP-868-041A Firmware 5.3.3

DEVICE INFORMATION

Device

Model Number	MTCAP-868-041A
Serial Number	21884112
Firmware	5.3.3
Current Time	10/14/2022 18:01:19
Up Time	00:09:16
WAN Transport	None
Current DNS	Not Acquired

LAN

Bridge	(br0)
MAC Address	00:08:00:4C:10:9D
IPv4 Address	192.168.2.1
Mask	255.255.255.0
DHCP State	Enabled
Lease Range	192.168.2.100-192.168.2.254
Interfaces	eth0

Ethernet	(eth0)
Bridge	br0
MAC Address	00:08:00:4C:10:9D

Lora

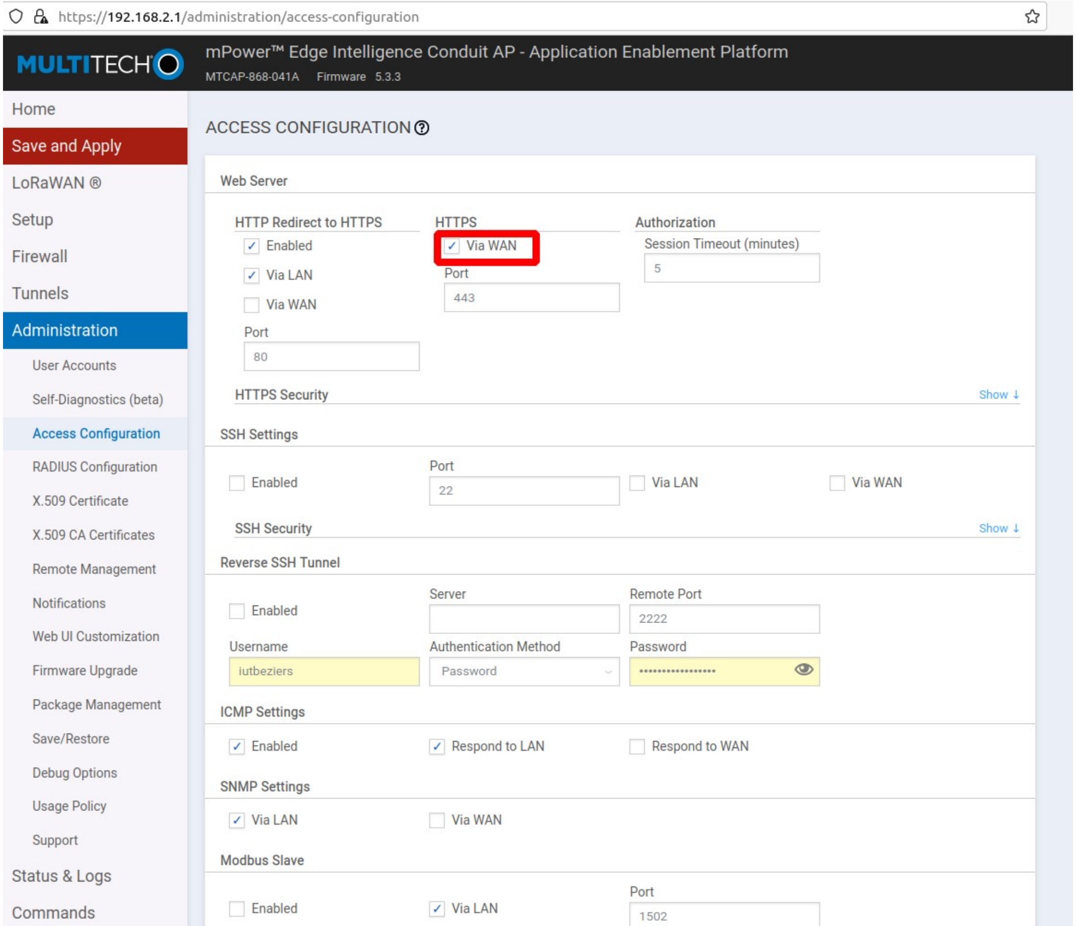
Frequency Band	868
EUI	00-80-00-00-00-02-1B-9F
Hardware	MTCAP-LORA-1.5

Accessory Cards

LORA	
Model Number	MTCAP-LORA-868
Hardware	MTCAP-LORA-1.5

Last updated: 5:04:31 PM

2.2. Définition de eth0 en client



ACCESS CONFIGURATION

Web Server

HTTP Redirect to HTTPS
☑ Enabled
☑ Via LAN
☐ Via WAN
Port: 80

HTTPS
☑ Via WAN
Port: 443

Authorization
Session Timeout (minutes): 5

HTTPS Security

SSH Settings
☐ Enabled
Port: 22
☐ Via LAN
☐ Via WAN

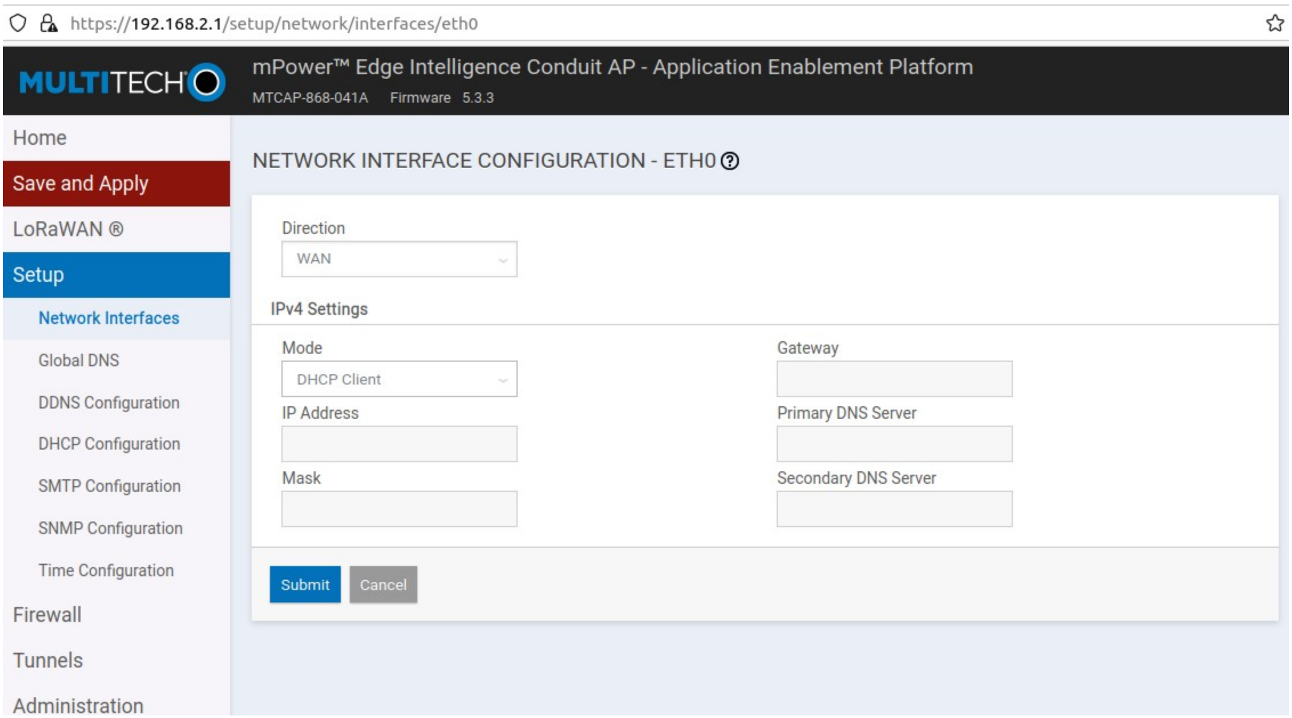
SSH Security

Reverse SSH Tunnel
☐ Enabled
Server:
Remote Port: 2222
Username: iutbeziers
Authentication Method: Password
Password: •••••••••••••••••

ICMP Settings
☑ Enabled
☑ Respond to LAN
☐ Respond to WAN

SNMP Settings
☑ Via LAN
☐ Via WAN

Modbus Slave
☐ Enabled
☑ Via LAN
Port: 1502



NETWORK INTERFACE CONFIGURATION - ETH0

Direction: WAN

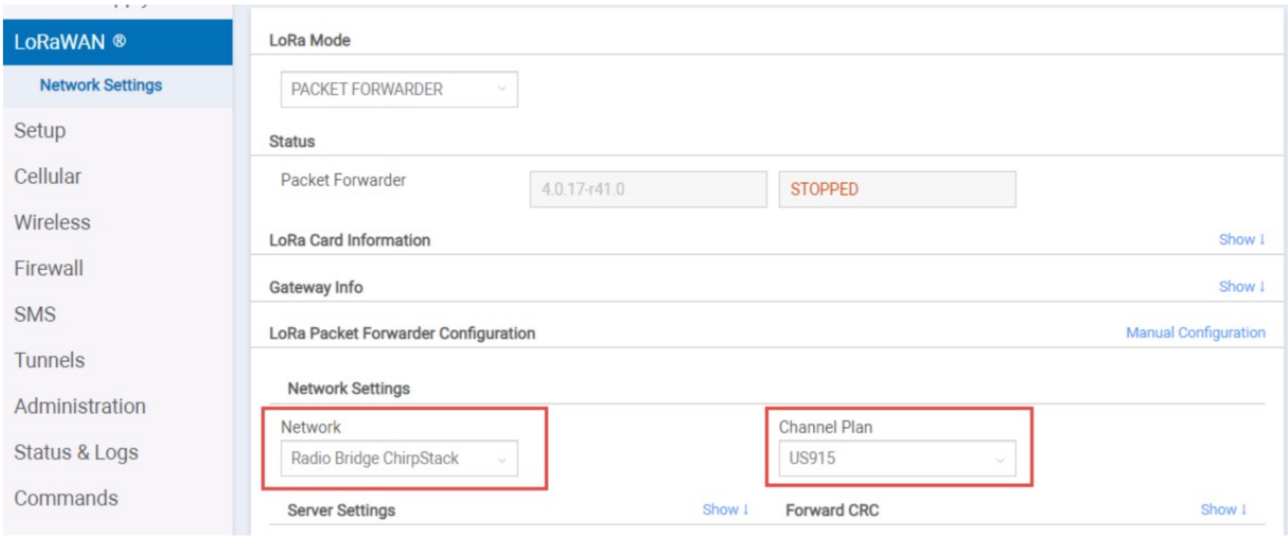
IPv4 Settings

Mode: DHCP Client
IP Address:
Mask:
Gateway:
Primary DNS Server:
Secondary DNS Server:

3. Lorawan

Dans Lorawan, j'ai sélectionné :

- mode : "Packet Forwarder"
- Network Setting : "The Things Networks" (et pas Radio Bridge ChirpStack comme indiqué ci-dessous). J'ai copié/collé depuis TTN l'URL du serveur Europe : "eu1.cloud.thethings.network" J'ai laissé les ports par défaut 1700/1700.
- Channel Plan : EU868: 868 (et pas US915 comme indiqué ci-dessous)



LoRa Mode

PACKET FORWARDER

Status

Packet Forwarder 4.0.17-r41.0 STOPPED

LoRa Card Information

Gateway Info

LoRa Packet Forwarder Configuration

Network Settings

Network: Radio Bridge ChirpStack

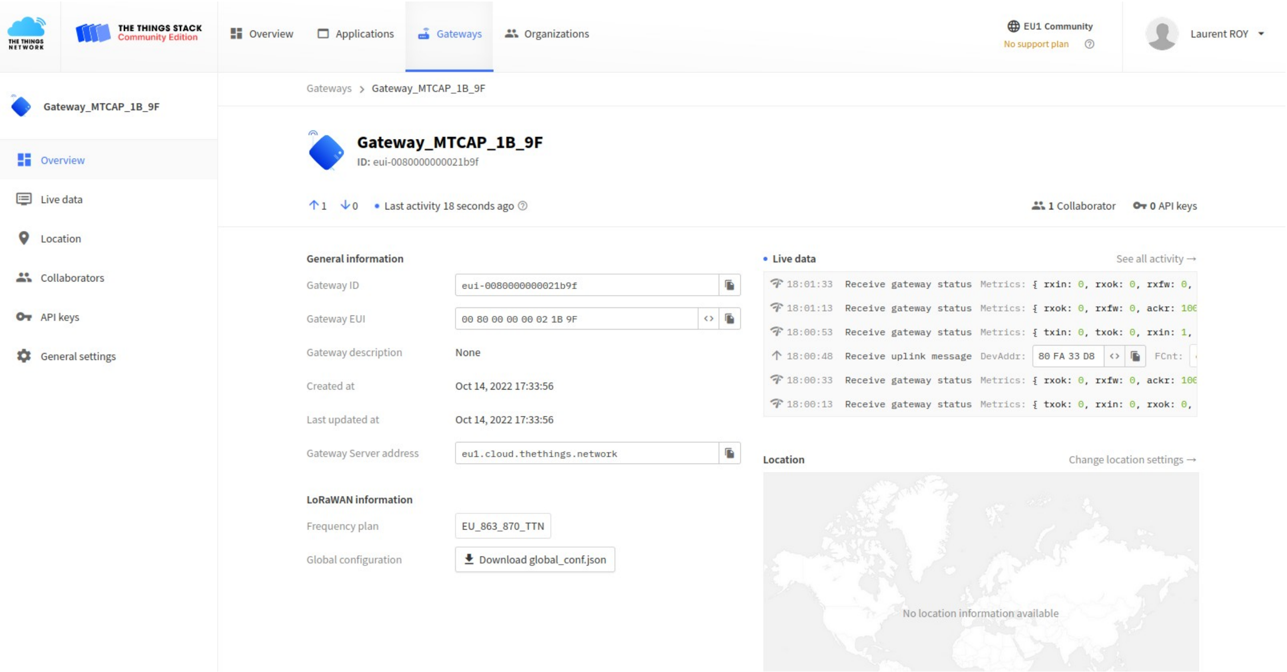
Channel Plan: US915

Server Settings

Forward CRC

4. Test Connexion

On parvient bien à voir la gateway sur TTN :



Gateway_MTCAP_1B_9F
ID: eui-0080000000021b9f

↑1 ↓0 • Last activity 18 seconds ago

1 Collaborator 0 API keys

General information

Gateway ID	eui-0080000000021b9f
Gateway EUI	00 80 00 00 00 02 1B 9F
Gateway description	None
Created at	Oct 14, 2022 17:33:56
Last updated at	Oct 14, 2022 17:33:56
Gateway Server address	eu1.cloud.thethings.network

LoRaWAN information

Frequency plan	EU_863_870_TTN
Global configuration	Download global_conf.json

• Live data

```
18:01:33  Receive gateway status  Metrics: { rxin: 0, rxok: 0, rxfw: 0,
18:01:13  Receive gateway status  Metrics: { rxok: 0, rxfw: 0, ackr: 10
18:00:53  Receive gateway status  Metrics: { txin: 0, txok: 0, rxin: 1,
18:00:48  Receive uplink message  DevAddr: 80 FA 33 D8  FCnt:
18:00:33  Receive gateway status  Metrics: { rxok: 0, rxfw: 0, ackr: 10
18:00:13  Receive gateway status  Metrics: { txok: 0, rxin: 0, rxok: 0,
```

Location

No location information available

Si on clique sur un évènement :

```
{
  "name": "gs.status.receive",
  "time": "2022-10-14T15:58:13.278211532Z",
  "identifiers": [
    {
      "gateway_ids": {
        "gateway_id": "eui-0080000000021b9f",
        "eui": "0080000000021B9F"
      }
    }
  ],
  "data": {
    "@type": "type.googleapis.com/ttn.lorawan.v3.GatewayStatus",
    "time": "2022-10-14T16:38:24Z",
    "versions": {
      "ttn-lw-gateway-server": "3.22.0-rc0-SNAPSHOT-98bce7902"
    },
    "ip": [
      "81.64.115.61"
    ],
    "metrics": {
      "txin": 0,
      "txok": 0,
      "rxin": 0,
      "rxok": 0,
      "rxfw": 0,
      "ackr": 100
    }
  },
  "correlation_ids": [
    "gs:conn:01GFBKDT2J99DS77CJEMAAND4N",
    "gs:status:01GFBKFARYPY2ZTJ4X594P11VG"
  ],
  "origin": "ip-10-100-12-248.eu-west-1.compute.internal",
  "context": {
    "tenant-id": "CgN0dG4="
  },
  "visibility": {
    "rights": [
      "RIGHT_GATEWAY_STATUS_READ"
    ]
  },
  "unique_id": "01GFBKFARYYQX3H5ANBY5J0MCD"
}
```

5. Raccordement d'un device

On met un pile (3.6V) dans le capteur.

On le configure avec l'appli sensor settings

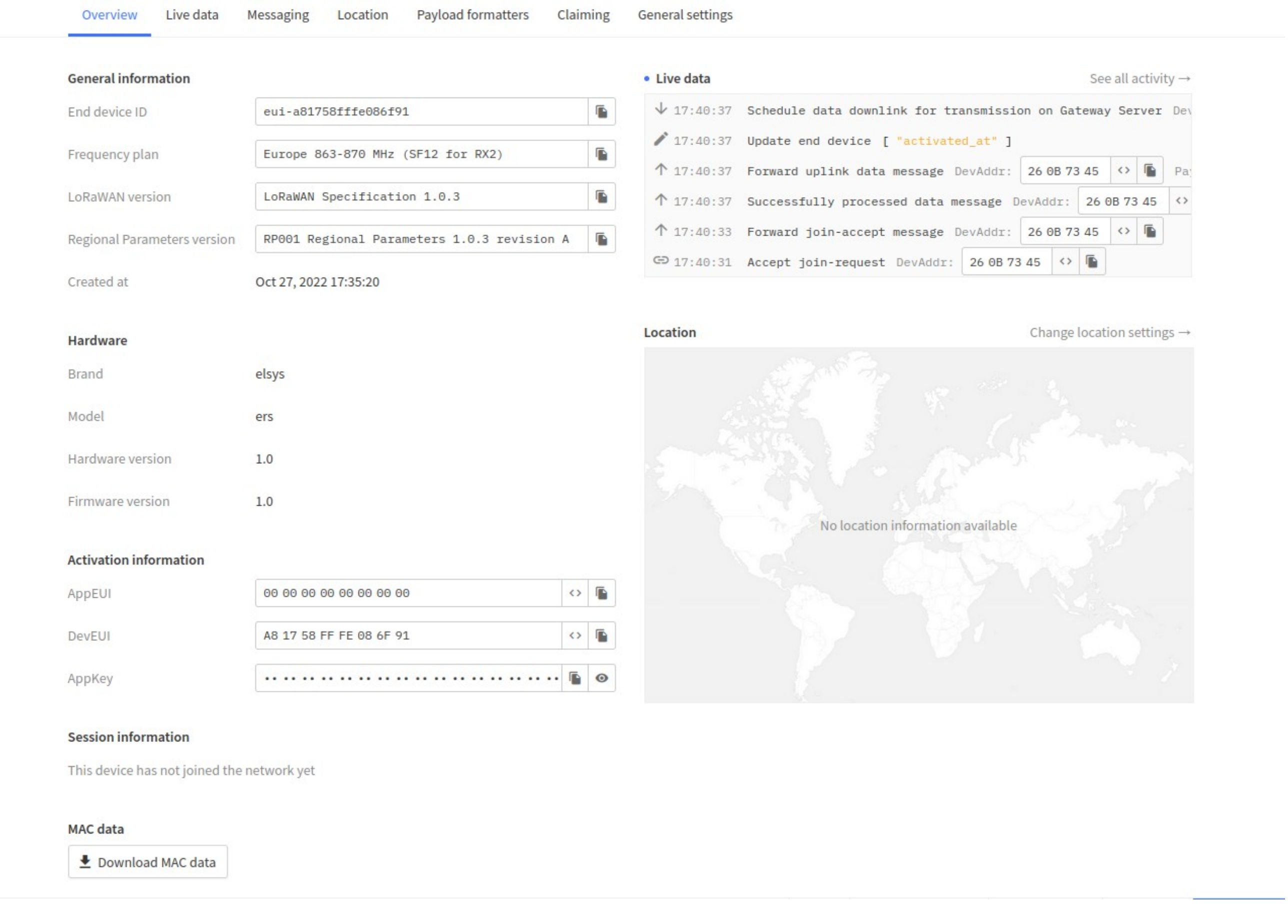
On crée une application dans TTN.

On ajoute un device :

Le code barre aurait été bien pratique, mais ne fonctionne pas.

Join Server TTN ? Comme on ne le connaît pas on met des 00...

On récup la app key et le dev eui par l'appli + mail (mais



General information

End device ID: eui-a81758fffe086f91

Frequency plan: Europe 863-870 MHz (SF12 for RX2)

LoRaWAN version: LoRaWAN Specification 1.0.3

Regional Parameters version: RP001 Regional Parameters 1.0.3 revision A

Created at: Oct 27, 2022 17:35:20

Hardware

Brand: elsys

Model: ers

Hardware version: 1.0

Firmware version: 1.0

Activation information

AppEUI: 00 00 00 00 00 00 00 00

DevEUI: A8 17 58 FF FE 08 6F 91

AppKey: •• •• •• •• •• •• •• •• •• •• •• •• •• •• •• ••

Session information

This device has not joined the network yet

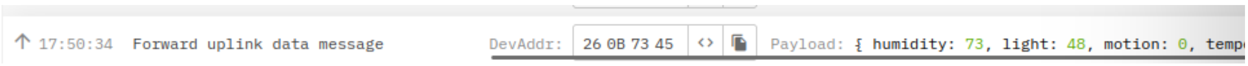
MAC data

Live data

```
17:40:37  Schedule data downlink for transmission on Gateway Server  Dev
17:40:37  Update end device [ "activated_at" ]
17:40:37  Forward uplink data message  DevAddr: 26 0B 73 45  Pa
17:40:37  Successfully processed data message  DevAddr: 26 0B 73 45
17:40:33  Forward join-accept message  DevAddr: 26 0B 73 45
17:40:31  Accept join-request  DevAddr: 26 0B 73 45
```

Location

Un message uplink reçu ::

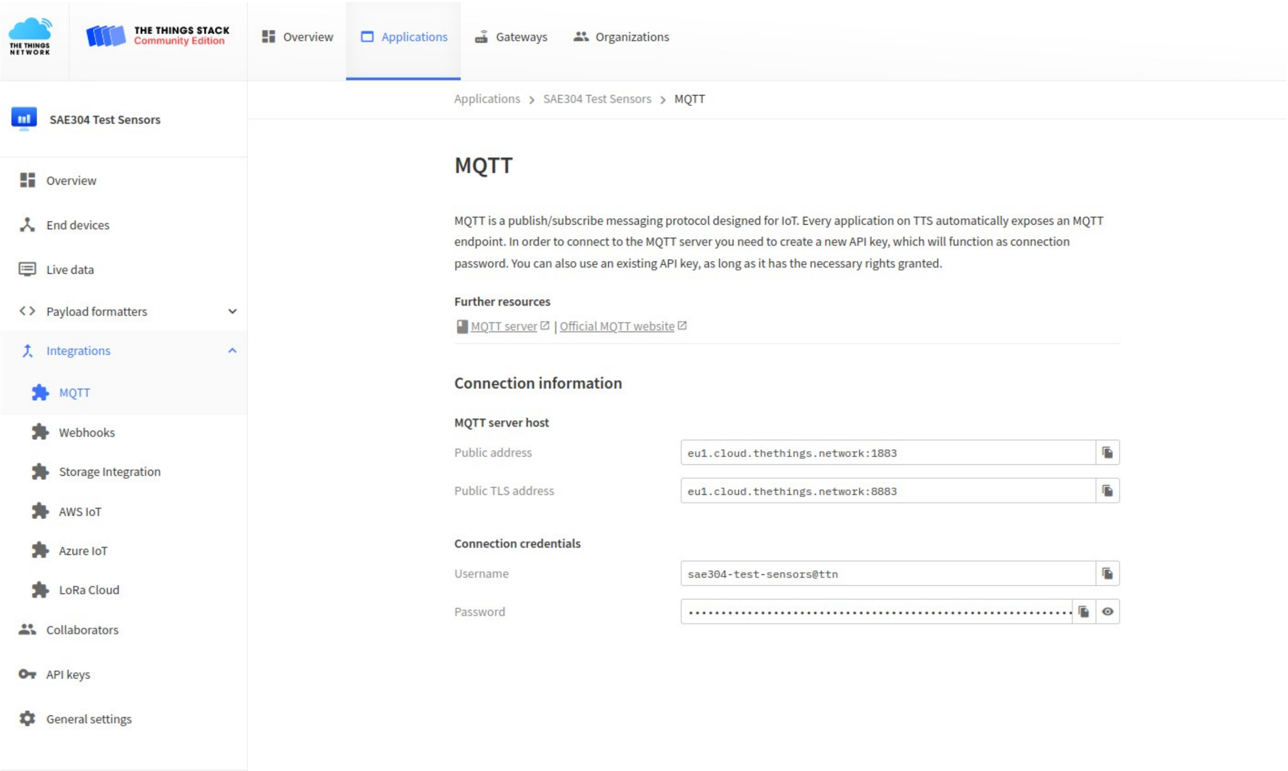


```
17:50:34  Forward uplink data message        DevAddr: 26 0B 73 45    Payload: { humidity: 73, light: 48, motion: 0, temp
```

```
{
  "name": "as.up.data.forward",
  "time": "2022-10-27T15:50:34.259701307Z",
  "identifiers": [
    {
      "device_ids": {
        "device_id": "eui-a81758fffe086f91",
        "application_ids": {
          "application_id": "sae304-test-sensors"
        },
        "dev_eui": "A81758FFFE086F91",
        "join_eui": "0000000000000000",
        "dev_addr": "260B7345"
      }
    }
  ],
  "data": {
    "@type": "type.googleapis.com/ttn.lorawan.v3.ApplicationUp",
    "end_device_ids": {
      "device_id": "eui-a81758fffe086f91",
      "application_ids": {
        "application_id": "sae304-test-sensors"
      },
      "dev_eui": "A81758FFFE086F91",
      "join_eui": "0000000000000000",
      "dev_addr": "260B7345"
    },
    "correlation_ids": [
      "as:up:01GGD26NGHNPK35Z9GSQ401NKC",
      "gs:conn:01GGCZDGV7DG5TYW1DW0JY41MN",
      "gs:up:host:01GGCZDGVDVDXJE8431XRJAWXC",
      "gs:uplink:01GGD26NA29C8S7P9YCK6VTC0S",
      "ns:uplink:01GGD26NA3YNP6TWQHKSAR6HP5",
      "rpc:/ttn.lorawan.v3.GsNs/HandleUplink:01GGD26NA39HNPECYJ50G77EZ0",
      "rpc:/ttn.lorawan.v3.NsAs/HandleUplink:01GGD26NGGVGGPPY7JJSE2H2NV"
    ],
    "received_at": "2022-10-27T15:50:34.257074068Z",
    "uplink_message": {
      "session_key_id": "AYQaGiUurNecjPU7CLFw+g==",
      "f_port": 5,
      "f_cnt": 11,
      "frm_payload": "AQD8AkkEADAFAAcOSg==",
      "decoded_payload": {
        "humidity": 73,
        "light": 48,
        "motion": 0,
        "temperature": 25.2,
        "vdd": 3658
      },
      "rx_metadata": [
        {
          "gateway_ids": {
            "gateway_id": "eui-0080000000021b9f",
            "eui": "0080000000021B9F"
          },
          "timestamp": 2933540627,
          "rssi": -36,
          "channel_rssi": -36,
          "snr": 10.5,
          "uplink_token": "CiIKIAoUZXVpLTAwODAwMDAwMDAwMjFiOWYSCACAAAAAAhufEJOO6fYKGgsIytHqmgYQi+2VGCC4xJalsFU=",
          "channel_index": 4,
          "received_at": "2022-10-27T15:50:34.030471387Z"
        }
      ],
      "settings": {
        "data_rate": {
          "lora": {
            "bandwidth": 125000,
            "spreading_factor": 7,
            "coding_rate": "4/5"
          }
        },
        "frequency": "867300000",
        "timestamp": 2933540627
      },
      "received_at": "2022-10-27T15:50:34.051296393Z",
      "consumed_airtime": "0.061696s",
      "version_ids": {
        "brand_id": "elsys",
        "model_id": "ers",
        "hardware_version": "1.0",
        "firmware_version": "1.0",
        "band_id": "EU_863_870"
      },
      "network_ids": {
        "net_id": "000013",
        "tenant_id": "ttn",
        "cluster_id": "eu1",
        "cluster_address": "eu1.cloud.thethings.network"
      }
    }
  },
  "correlation_ids": [
    "as:up:01GGD26NGHNPK35Z9GSQ401NKC",
    "gs:conn:01GGCZDGV7DG5TYW1DW0JY41MN",
    "gs:up:host:01GGCZDGVDVDXJE8431XRJAWXC",
    "gs:uplink:01GGD26NA29C8S7P9YCK6VTC0S",
    "ns:uplink:01GGD26NA3YNP6TWQHKSAR6HP5",
    "rpc:/ttn.lorawan.v3.GsNs/HandleUplink:01GGD26NA39HNPECYJ50G77EZ0",
    "rpc:/ttn.lorawan.v3.NsAs/HandleUplink:01GGD26NGGVGGPPY7JJSE2H2NV"
  ],
  "origin": "ip-10-100-4-226.eu-west-1.compute.internal",
  "context": {
    "tenant-id": "CgN0dG4="
  },
  "visibility": {
    "rights": [
      "RIGHT_APPLICATION_TRAFFIC_READ"
    ]
  },
  "unique_id": "01GGD26NGKM8GHS807F7AZ3ZHP"
}
```

6. Intégration

On veut maintenant intégrer les données dans telegram. On va utiliser un protocole MQTT.



MQTT

MQTT is a publish/subscribe messaging protocol designed for IoT. Every application on TTS automatically exposes an MQTT endpoint. In order to connect to the MQTT server you need to create a new API key, which will function as connection password. You can also use an existing API key, as long as it has the necessary rights granted.

Further resources

MQTT server | Official MQTT website

Connection information

MQTT server host

Public address: eu1.cloud.thethings.network:1883

Public TLS address: eu1.cloud.thethings.network:8883

Connection credentials

Username: sae304-test-sensors@ttn

Password: ••

Depuis la même page on a généré une clef

Username : sae304-test-sensors@ttn

NNSXS.NCXZ7TGZ6NUEGEWMQ2T7II4QKDK3L4VKRSA5DGQ.O7OJRXNKGF2ACMBTB3YCPXULQHQQIA2ZTSIAYFFFNCXMDSRNSUKQ

7. Telegraf / InfluxDB / Grafana

On va essayer de se passer de docker. On suit principalement cette source, qui n'est pas spécifique à l'IoT mais qui permet de mettre en place une serveur 'TIG' orienté supervision informatique. On l'adaptera ensuite pour l'IoT

https://www.howtoforge.com/how-to-install-tig-stack-telegraf-influxdb-and-grafana-on-ubuntu-22-04/

7.1. InfluxDB et Telegraf

On suit les étapes du lien précédent.

```
root@IOT3:~# influx setup
> Welcome to InfluxDB 2.0!
? Please type your primary username lr
? Please type your password  o4E660dS&a4p
? Please type your password again  o4E660dS&a4p
? Please type your primary organization name iutbeziers
? Please type your primary bucket name telegraf
? Please type your retention period in hours, or 0 for infinite 0
? Setup with these parameters?
  Username:          lr
  Organization:      iutbeziers
  Bucket:            telegraf
  Retention Period:  infinite
```

telegraf token :

x-kAVNJAfjVuvIVCqJ6ou5akueE_OuDZoA4OOQx8r2hCMvxwjYdVtyII8CcspxfkmEWiVlvAtDfWLqyXih1Mcw==

Remarque : dans telegraf, il faut attendre que les données de cpu arrivent (une 10aine de min.)

7.2. Grafana

On suit les étapes du lien précédent.

a) Compte

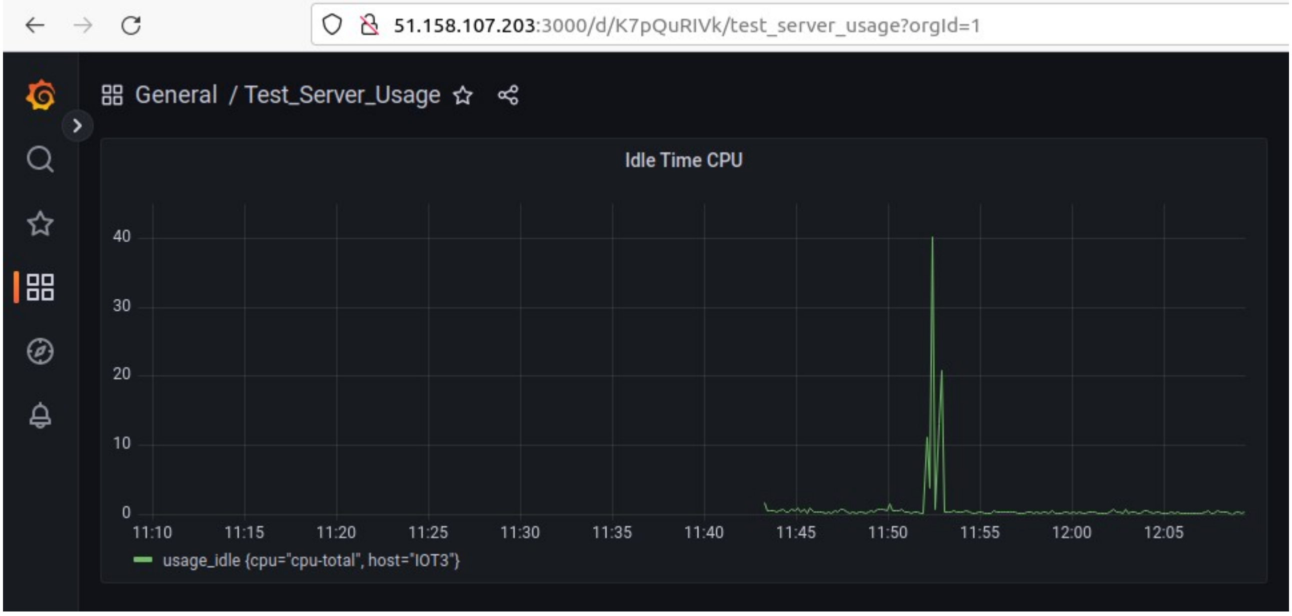
password admin grafana : H93l7ri4B

b) Dashboard

script à rentrer dans le dashboard pour récupérer les données de la CPU

```
from(bucket: "telegraf")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["_measurement"] == "cpu")
  |> filter(fn: (r) => r["_field"] == "usage_idle")
  |> filter(fn: (r) => r["cpu"] == "cpu-total")
  |> filter(fn: (r) => r["host"] == "IOT4")
  |> aggregateWindow(every: v.windowPeriod, fn: mean, createEmpty: false)
  |> map(fn: (r) => ({ r with _value: r._value * -1.0 + 100.0 }))
  |> toFloat()
  |> yield(name: "mean")
```

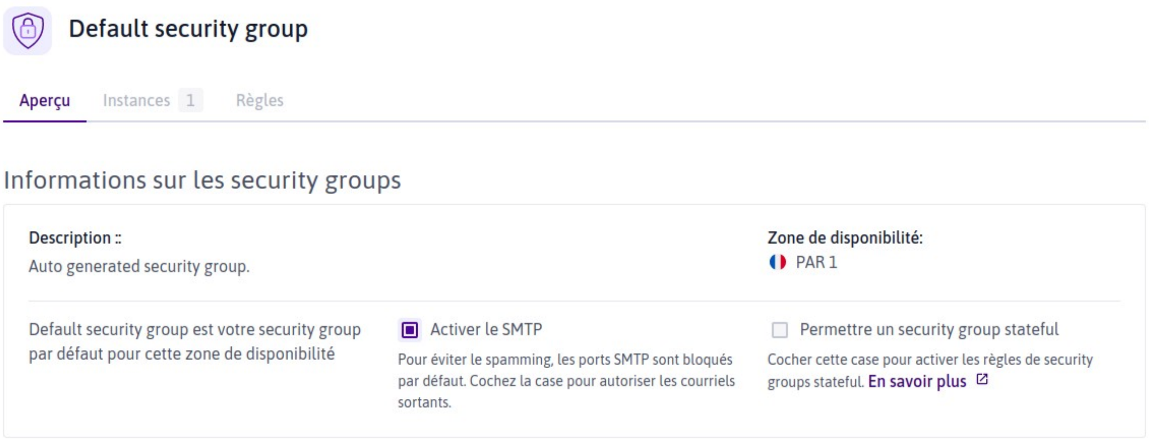
Résultat :



51.158.107.203:3000/d/K7pQuRIVk/test_server_usage?orgId=1

c) Alertes (Envoi de mails)

On active SMTP dans le security group



Default security group

Informations sur les security groups

Description ::
Auto generated security group.

Zone de disponibilité:
PAR 1

Default security group est votre security group par défaut pour cette zone de disponibilité

☑ Activer le SMTP
Pour éviter le spamming, les ports SMTP sont bloqués par défaut. Cochez la case pour autoriser les courriels sortants.

□ Permettre un security group stateful
Cocher cette case pour activer les règles de security groups stateful. En savoir plus

Pour faire une notification, on va essayer d'utiliser le gmail walt.white69230@gmail.com

mdp : 13P19#D4

mais nécessite un app password :

https://stackoverflow.com/questions/72478573/how-to-send-an-email-using-python-after-googles-policy-update-on-not-allowing-j

Mot de passe application pour le compte :

sxteopzqhcsornwk

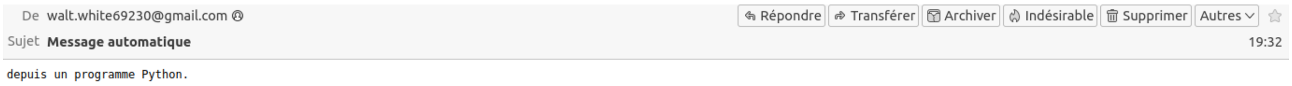
Programme python :

```
root@IOT3:~# cat env_mail.py
import smtplib, ssl

port = 587  # For starttls
smtp_server = "smtp.gmail.com"
sender_email = "walt.white69230@gmail.com"  # Enter your address
receiver_email = "lroy.perso@gmail.com"  # Enter receiver address
password = input("Entrer le mot de passe :")
message = """\
Subject: Message automatique

depuis un programme Python."""


context = ssl.create_default_context()
with smtplib.SMTP(smtp_server, port) as server:
    server.ehlo()  # Can be omitted
    server.starttls(context=context)
    server.ehlo()  # Can be omitted
    server.login(sender_email, password)
    server.sendmail(sender_email, receiver_email, message)
```



De walt.white69230@gmail.com
Sujet Message automatique
19:32

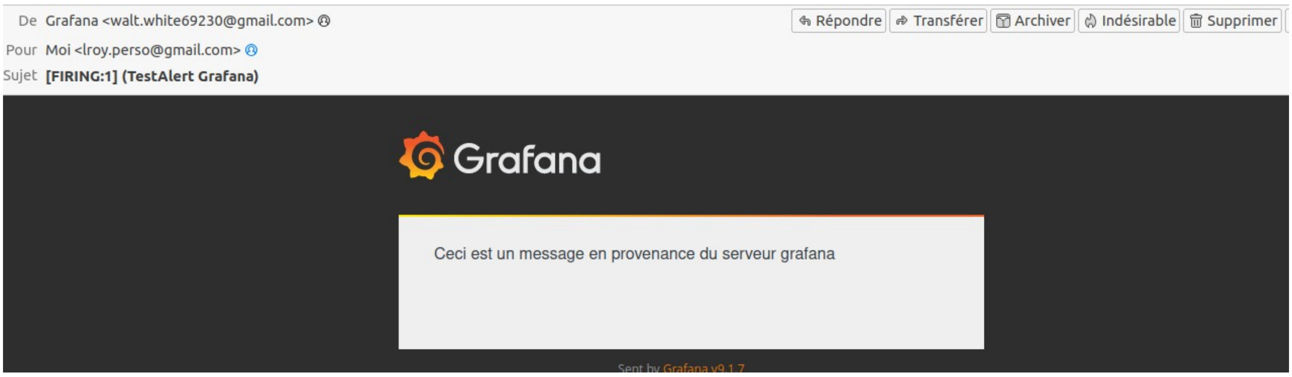
```
depuis un programme Python.
```

Remarque : pas d'accent dans le message (ascii non étendu!)

```
root@IOT3:~# sudo nano /etc/grafana/grafana.ini
```

```
#################################### SMTP / Emailing ##########################
[smtp]
enabled = true
host = smtp.gmail.com:587
user = walt.white69230@gmail.com
# If the password contains # or ; you have to wrap it with triple quotes. Ex
"""#password;"""
password = "sxteopzqhcsornwk"
;cert_file =
;key_file =
;skip_verify = false
from_address = walt.white69230@gmail.com
from_name = Grafana
# EHLO identity in SMTP dialog (defaults to instance_name)
;ehlo_identity = dashboard.example.com
# SMTP startTLS policy (defaults to 'OpportunisticStartTLS')
;startTLS_policy = NoStartTLS
```

Cela fonctionne :



De Grafana <walt.white69230@gmail.com>

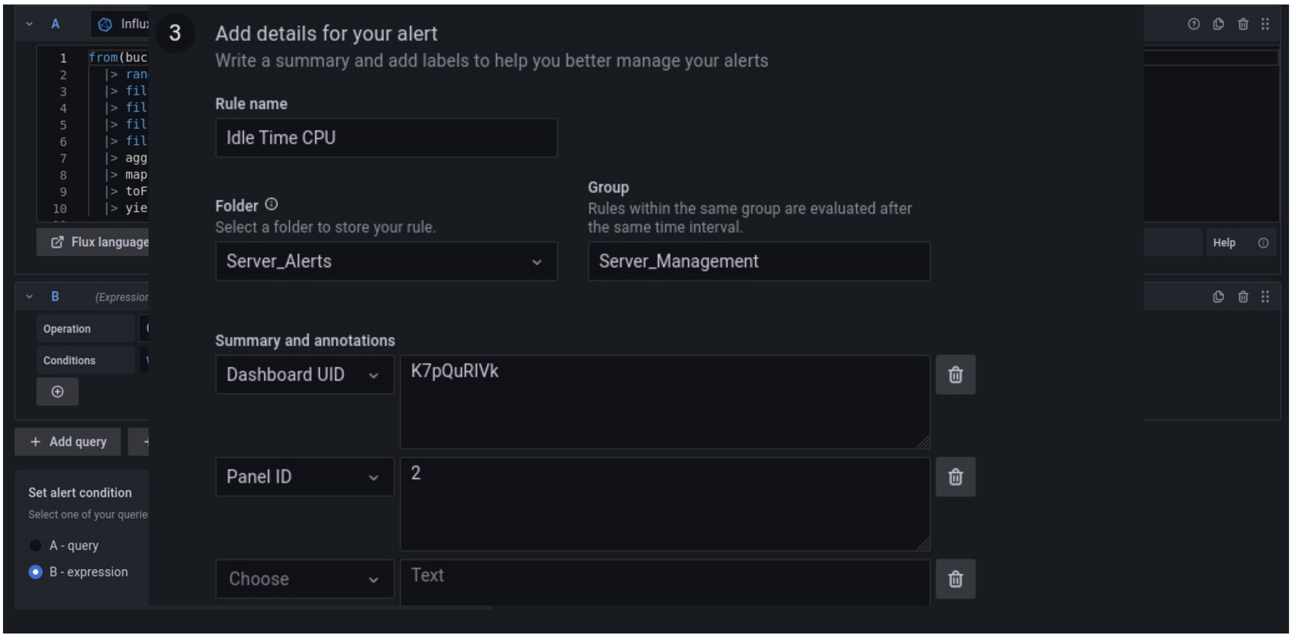
Pour Moi <lroy.perso@gmail.com>

Sujet **[FIRING:1] (TestAlert Grafana)**

Ceci est un message en provenance du serveur grafana

Sent by Grafana v9.1.7

On définit une alerte sur la cpu



```
from(buc
  |> ran
  |> fil
  |> fil
  |> fil
  |> fil
  |> agg
  |> map
  |> toF
  |> yie
```

3 Add details for your alert

Write a summary and add labels to help you better manage your alerts

Rule name: Idle Time CPU

Folder: Server_Alerts

Select a folder to store your rule.

Group: Server_Management

Rules within the same group are evaluated after the same time interval.

Summary and annotations

Dashboard UID: K7pQuRIVk

Panel ID: 2

Choose: Text

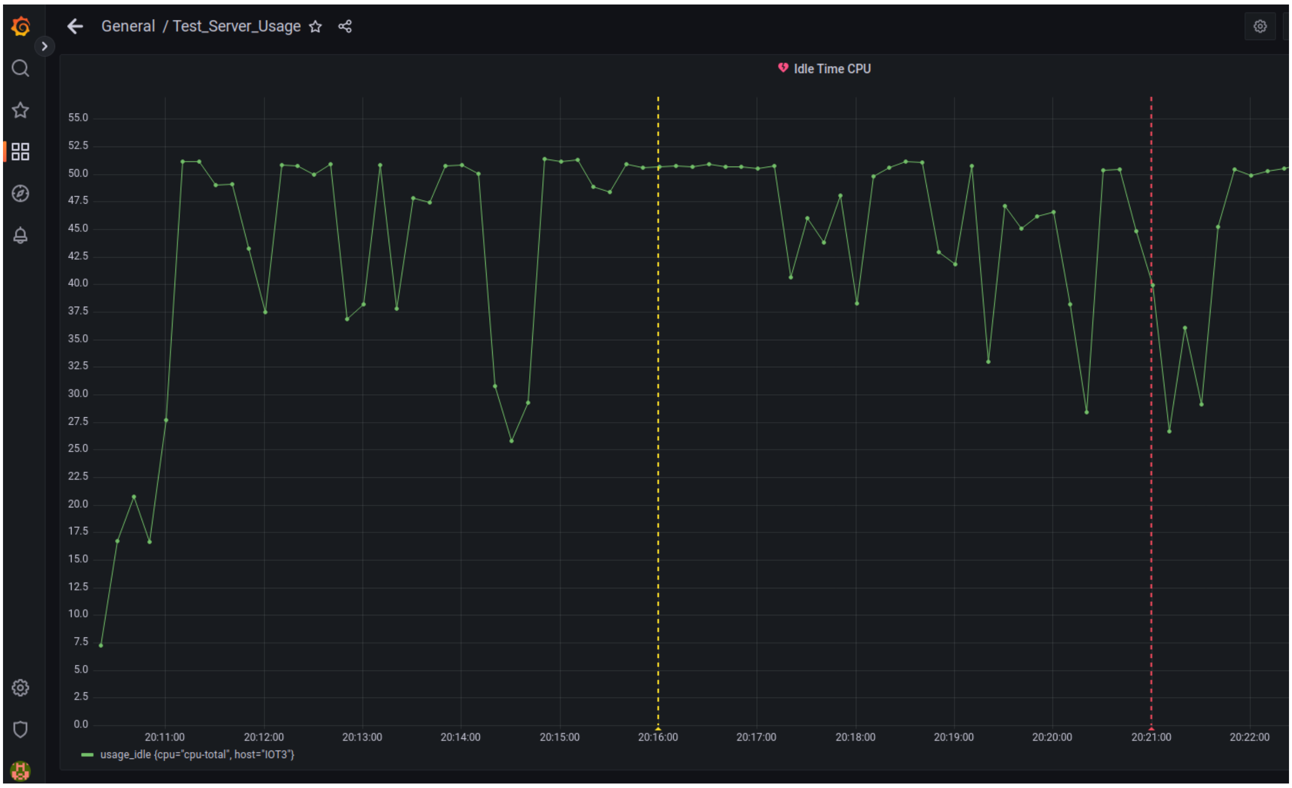
Set alert condition

Select one of your querie

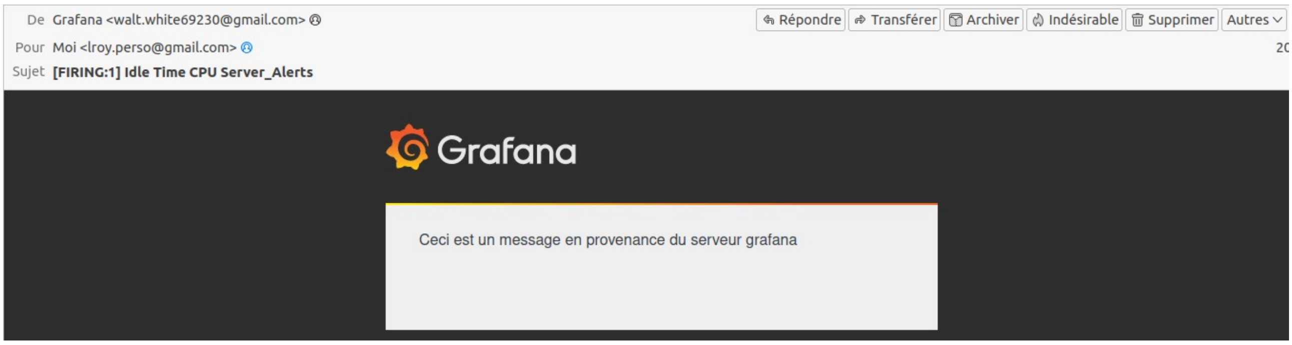
○ A - query

● B - expression

On la fait mouliner avec un calcul de factoriels



On obtient bien un message (pas super personnalisé...)



De Grafana <walt.white69230@gmail.com>

Pour Moi <lroy.perso@gmail.com>

Sujet **[FIRING:1] Idle Time CPU Server_Alerts**

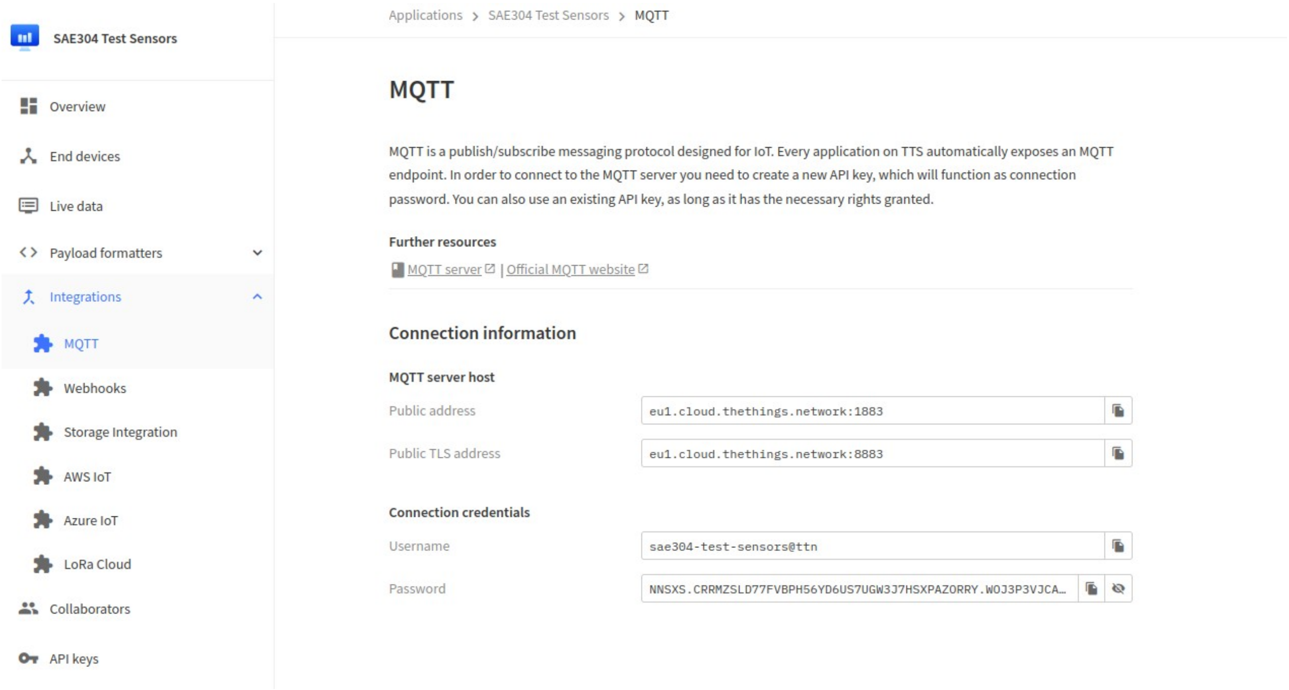
Ceci est un message en provenance du serveur grafana

ToDo : Personnaliser le message

7.3. Adaptation à l'IOT

Les données sont présentes dans TTN, on aimerait bien les avoir à disposition dans telegraf et grafana. On va se baser pour cela sur les indications qui figurent ici :

https://www.influxdata.com/blog/revisiting-things-network-connecting-things-network-influxdb/



MQTT

MQTT is a publish/subscribe messaging protocol designed for IoT. Every application on TTS automatically exposes an MQTT endpoint. In order to connect to the MQTT server you need to create a new API key, which will function as connection password. You can also use an existing API key, as long as it has the necessary rights granted.

Further resources

MQTT server | Official MQTT website

Connection information

MQTT server host

Public address: eu1.cloud.thethings.network:1883

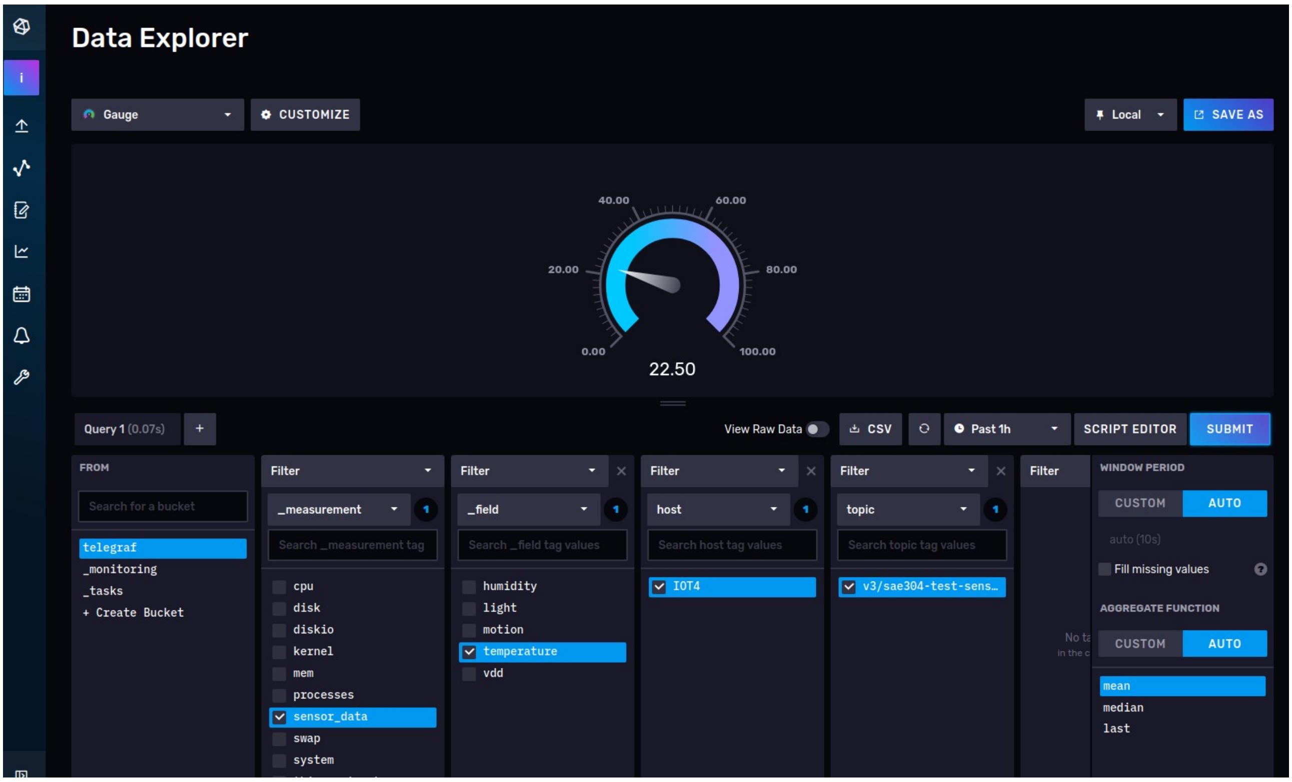
Public TLS address: eu1.cloud.thethings.network:8883

Connection credentials

Username: sae304-test-sensors@ttn

Password: NNSXS.CRRMZSLD77FVBPH56YD6US7UGW3J7HSXPAZORRY.WOJ3P3VJCA…

On ajoute les inputs de ttn dans telegraf :



FROM: telegraf
_monitoring
_tasks
+ Create Bucket

Filter: _measurement 1
□ cpu
□ disk
□ diskio
□ kernel
□ mem
□ processes
☑ sensor_data
□ swap
□ system

Filter: _field 1
□ humidity
□ light
□ motion
☑ temperature
□ vdd

Filter: host 1
☑ IOT4

Filter: topic 1
☑ v3/sae304-test-sens…

Filter

WINDOW PERIOD
CUSTOM AUTO
auto (10s)
□ Fill missing values

AGGREGATE FUNCTION
CUSTOM AUTO
mean
median
last

En partant de ce fichier :
https://github.com/InfluxCommunity/plant_buddy/blob/master/microcontroller/thing_network/telegraf-mqtt-thing-network.conf

On adapte le INPUT PLUGINS à notre situation

```
###############################################################################
#                            INPUT PLUGINS                                    #
###############################################################################
            #THING NETWORK META DATA#
            [[inputs.mqtt_consumer]]
              alias = "thing_network_consumer"
              name_override = "thing_network"
              servers = ["tcp://eu1.cloud.thethings.network:1883"]
              topics = ["#"]
              max_undelivered_messages = 1


                username = "sae304-test-sensors@ttn"
                password = 
"NNSXS.CRRMZSLD77FVBPH56YD6US7UGW3J7HSXPAZORRY.WOJ3P3VJCADEW3MQO64FEBABOLBIHDMIW
FOELWN5P4GVBT5V2HIA"
                data_format = "json_v2"


              [[inputs.mqtt_consumer.json_v2]]
                [[inputs.mqtt_consumer.json_v2.tag]]
                path = "@this.end_device_ids.device_id"

              [[inputs.mqtt_consumer.json_v2.object]]
                path = "end_device_ids"
                disable_prepend_keys = true

              [[inputs.mqtt_consumer.json_v2.object]]
                path = "uplink_message"
                disable_prepend_keys = true
                excluded_keys = ["time", "timestamp"]

              [[inputs.mqtt_consumer.json_v2.object]]
                path = "uplink_message.rx_metadata"
                disable_prepend_keys = true
                excluded_keys = ["time", "timestamp"]





#PLANT BUDDY DATA#
[[inputs.mqtt_consumer]]
alias = "thing_network_consumer2"
name_override = "sensor_data"



  servers = ["tcp://eu1.cloud.thethings.network:1883"]
  topics = ["#"]


  username = "sae304-test-sensors@ttn"
  password = 
"NNSXS.CRRMZSLD77FVBPH56YD6US7UGW3J7HSXPAZORRY.WOJ3P3VJCADEW3MQO64FEBABOLBIHDMIW
FOELWN5P4GVBT5V2HIA"
  data_format = "json_v2"
```

```
 [[inputs.mqtt_consumer.json_v2]]
 [[inputs.mqtt_consumer.json_v2.object]]
  path = "@this.uplink_message.decoded_payload"
  disable_prepend_keys = true
```

et on le copie dans le /etc/telegraf/telegraf.conf de notre serveur IoT sur scaleway.

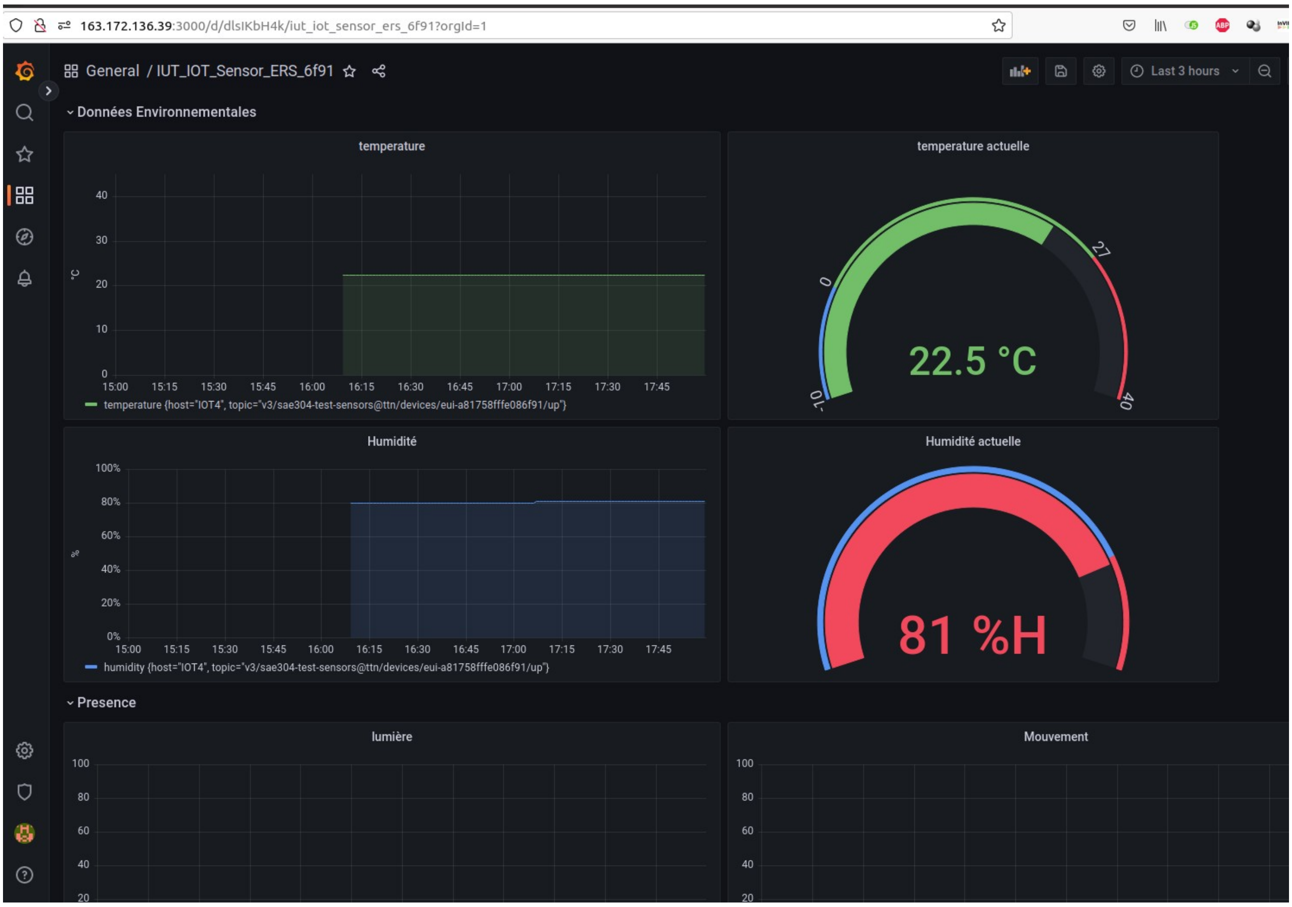
```
#nano telegraf.conf
#service telegraf stop
#service telegraf start
```

On va maintenant essayer de récupérer les données dans grafana

On crée un nouveau Dashboard

```
from(bucket: "telegraf")
|> range(start: v.timeRangeStart, stop: v.timeRangeStop)
|> filter(fn: (r) => r["_measurement"] == "sensor_data")
|> filter(fn: (r) => r["_field"] == "temperature")
|> filter(fn: (r) => r["host"] == "IOT4")
|> aggregateWindow(every: v.windowPeriod, fn: mean, createEmpty: false)
```

Cela fonctionne bien :


163.172.136.39:3000/d/dlsIKbH4k/iut_iot_sensor_ers_6f91?orgId=1
General / IUT_IOT_Sensor_ERS_6f91
Last 3 hours
Données Environnementales
temperature
40
30
20
10
0
°C
15:00
15:15
15:30
15:45
16:00
16:15
16:30
16:45
17:00
17:15
17:30
17:45
temperature {host="IOT4", topic="v3/sae304-test-sensors@ttn/devices/eui-a81758fffe086f91/up"}
temperature actuelle
0
27
-10
40
22.5 °C
Humidité
100%
80%
60%
40%
20%
0%
%
15:00
15:15
15:30
15:45
16:00
16:15
16:30
16:45
17:00
17:15
17:30
17:45
humidity {host="IOT4", topic="v3/sae304-test-sensors@ttn/devices/eui-a81758fffe086f91/up"}
Humidité actuelle
81 %H
Presence
lumière
100
80
60
40
20
Mouvement
100
80
60
40
20

Par contre on aimerait bien pouvoir partager le dashboard : pour cela plusieurs options :

On peut le faire publiquement, mais cela peut engendrer un trafic important

On peut le faire en utilisant un user viewer

on clique sur l'icone en forme de bouclier en bas à gauche. On crée un user

- name : visitor
- username : visitor@iut_iot
- pwd : iutbeziers

Si on partage le dashboard avec le bouton share, la personne qui a le lien et les identifiants ci-dessus peut bien suivre l'évolution des différentes mesures partout dans le monde.

TODO : trier les données :

https://community.influxdata.com/t/bucket-tag-to-direct-metrics-to-different-buckets/19312

https://docs.influxdata.com/telegraf/v1.18/administration/configuration/#measurement-filtering

https://www.youtube.com/watch?v=R3DnObs_OKA

8. Sources

https://hackmd.io/@pmanzoni/BkkbnmQ2H

Tutorial Multitech pour la configuration de la passerelle mtcap :

https://www.multitech.com/documents/publications/training/S000812--Configuring-mDot-w-MTACP-using-LoRa-App-Note.pdf

Documentation de la passerelle multitech mtcap-868-041a :
https://www.multitech.com/documents/publications/user-guides/S000793-MTCAP-L4E1-868-041.pdf

Pour mettre en place le serveur TIG

https://www.howtoforge.com/how-to-install-tig-stack-telegraf-influxdb-and-grafana-on-ubuntu-22-04/