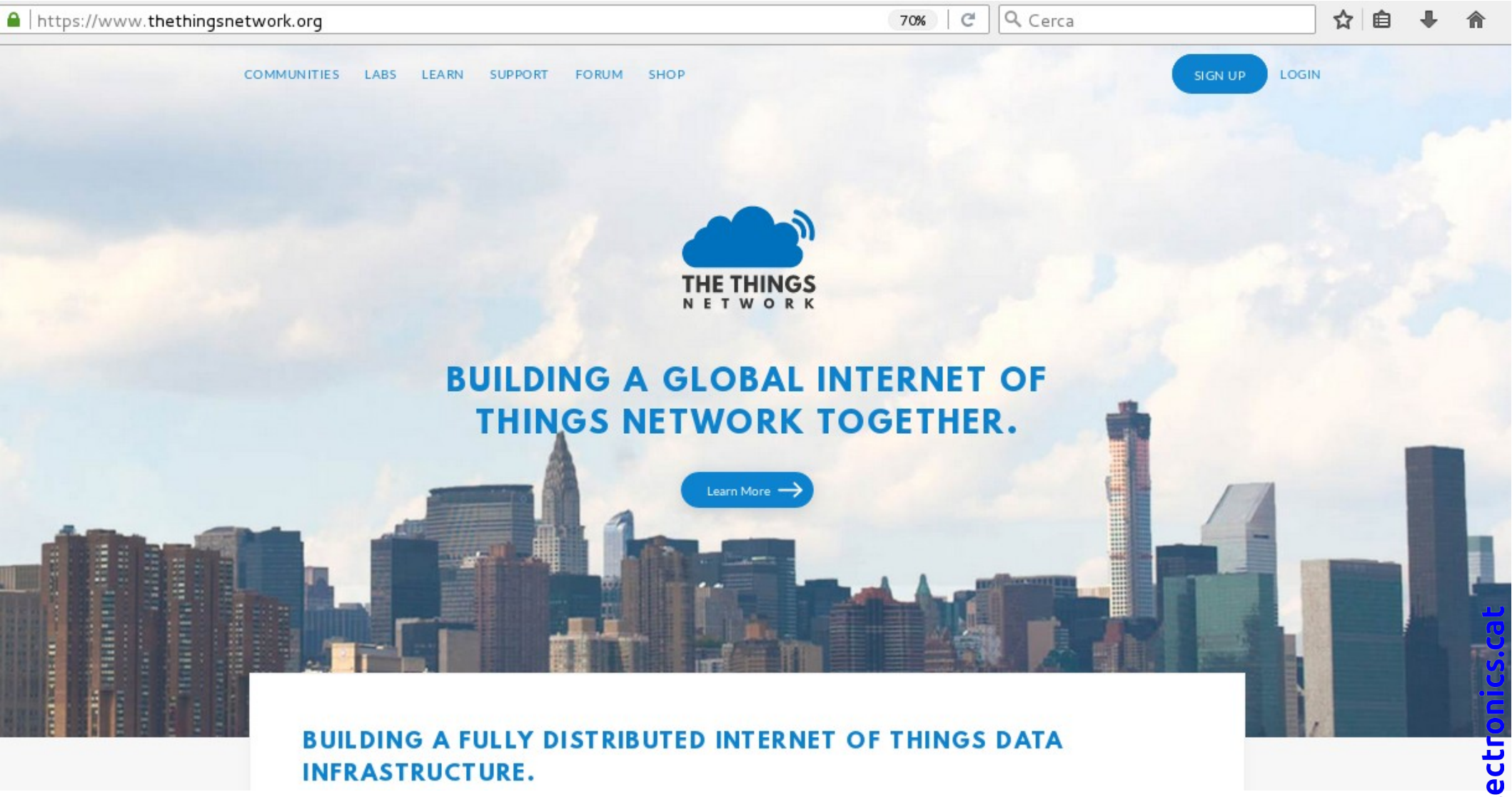
LoRaWAN™

The Things Network



https://www.thethingsnetwork.org/

La cònsola de The Things Network

https://console.thethingsnetwork.org

Cerca

THE THINGS NETWORK CONSOLE COMMUNITY EDITION

Applications Gateways Support

Welcome to The Things Network Console.

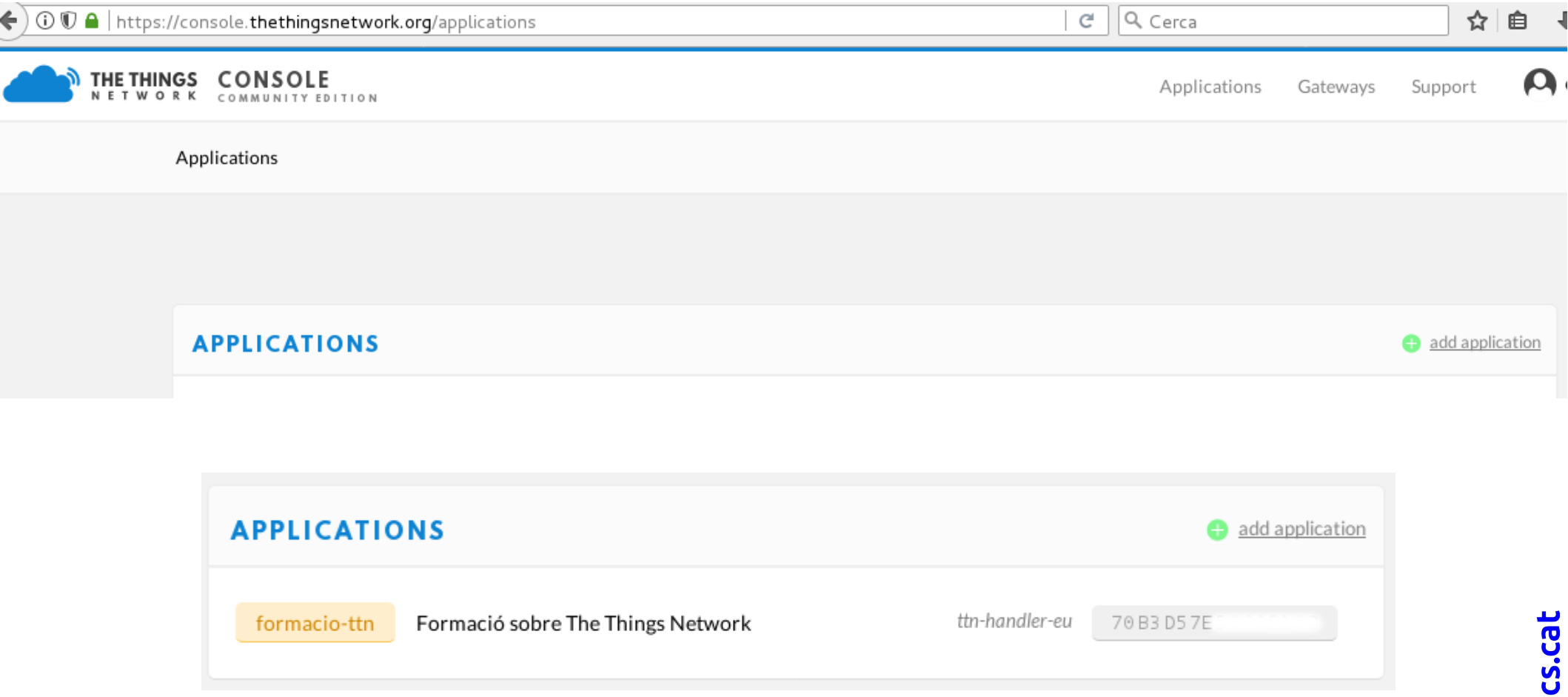
This is where the magic happens. Here you can work with your data. Register applications, devices and gateways, manage your integrations, collaborators and settings.

APPLICATIONS

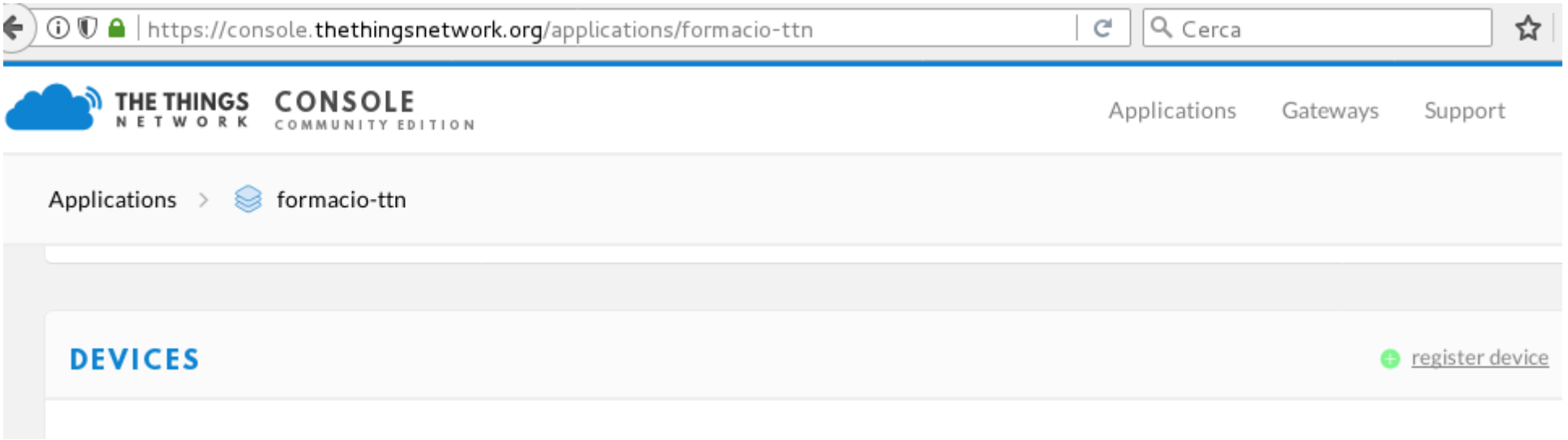
GATEWAYS

https://www.thethingsnetwork.org/

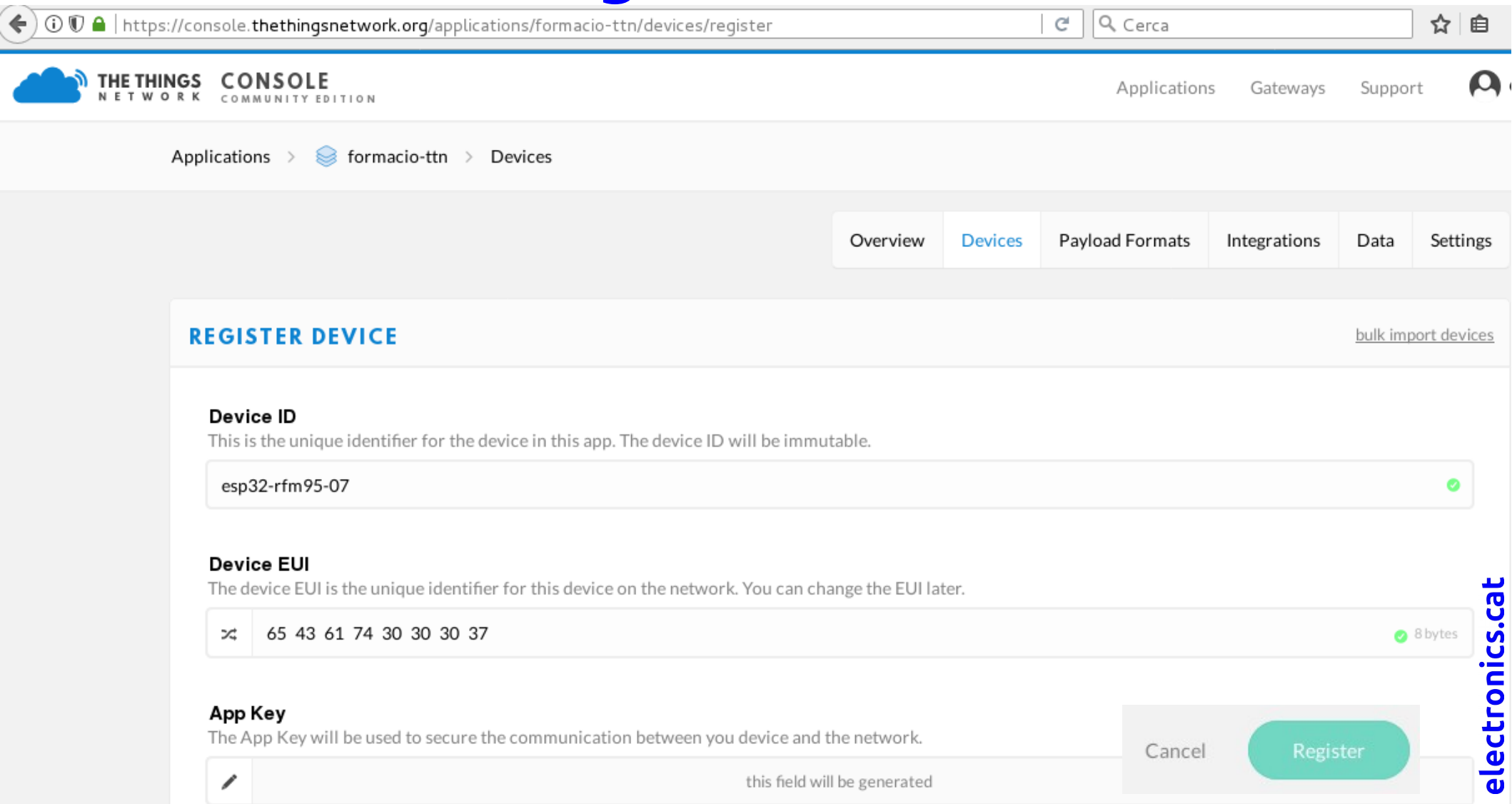
Nova aplicació a The Things Network



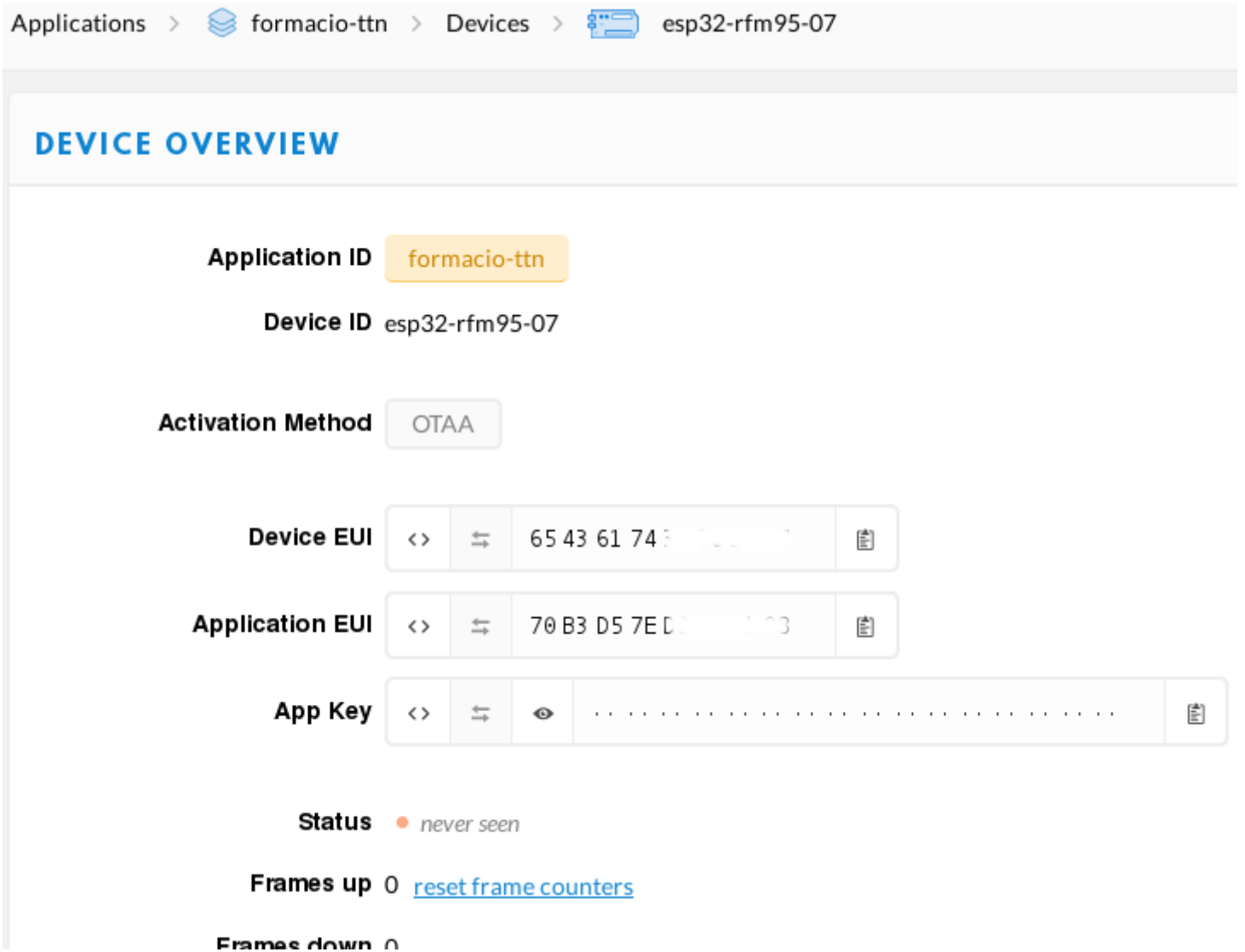
Registre de dispositiu a l'aplicació a The Things Network



Registre de dispositiu a l'aplicació a The Things Network (TTN)



Registre de dispositiu a l'aplicació a TTN

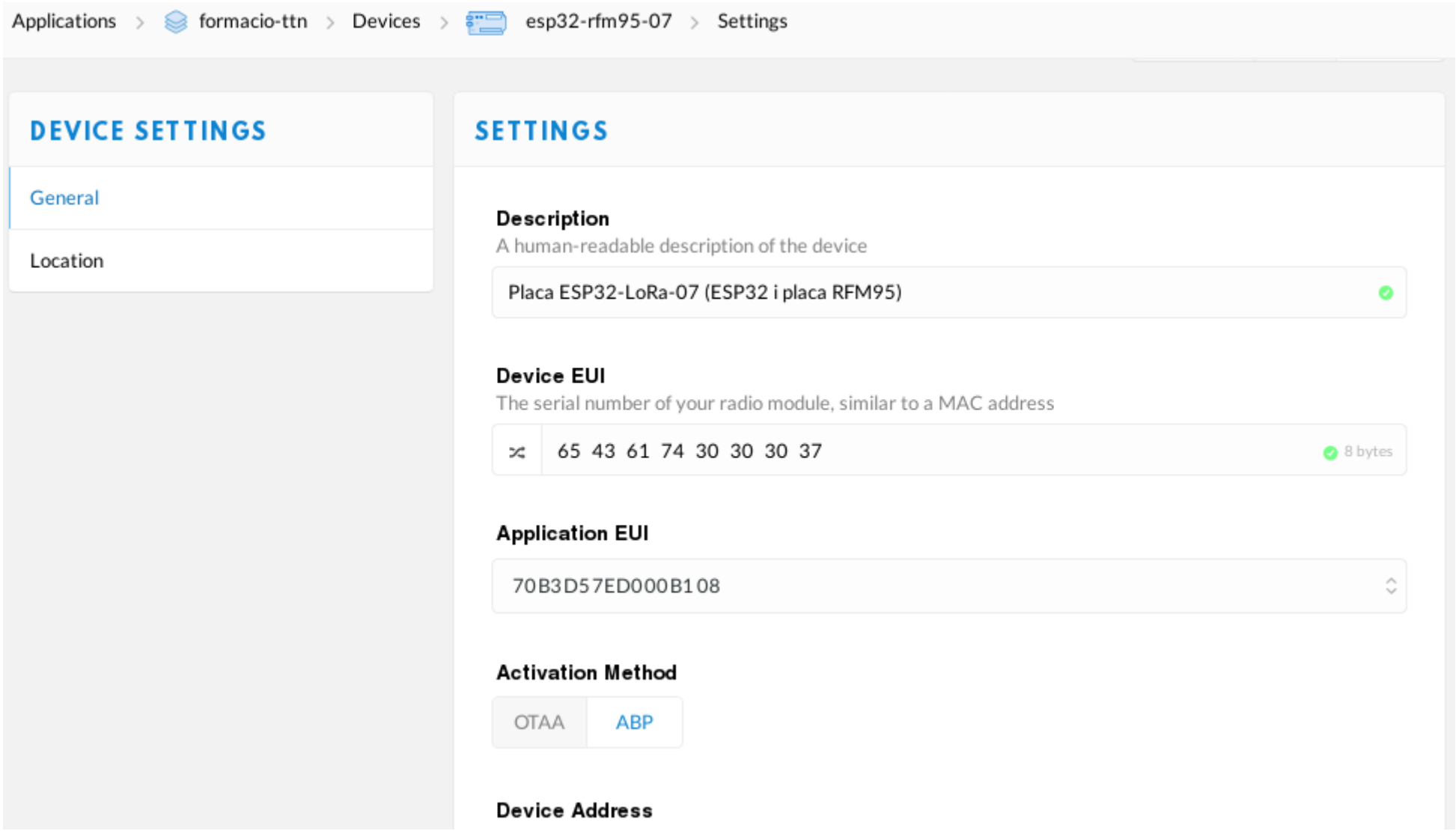


OTAA i ABP

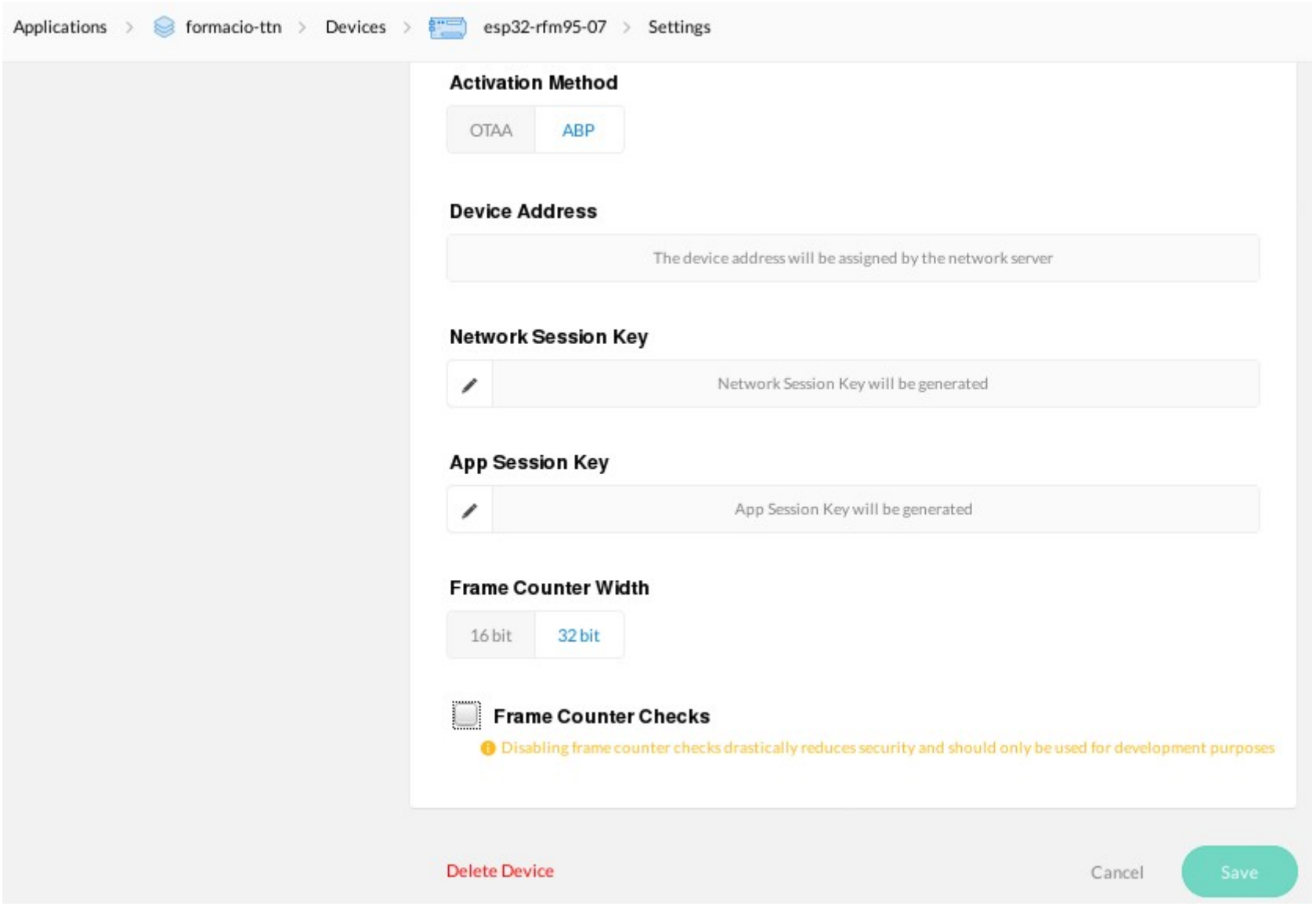
OTAA (Over The Air Authentication): Aquest mètode consisteix en que un node prova de connectar-se a una passarel·la LoRaWAN emprant les claus **APPEUI** i **APPKEY**. Si les claus són correctes, la passarel·la contestarà al node amb un missatge d'acceptació (join) i des d'aquest moment el node pot rebre i trametre des de la passarel·la.

ABP (Authentication By Personalisation): L'autenticació per personalització consisteix en que les claus de xifratge es configuren manualment al dispositiu, permetent la comunicació directa a les passarel·les LoRaWAN sense necessitat de demanar permís.

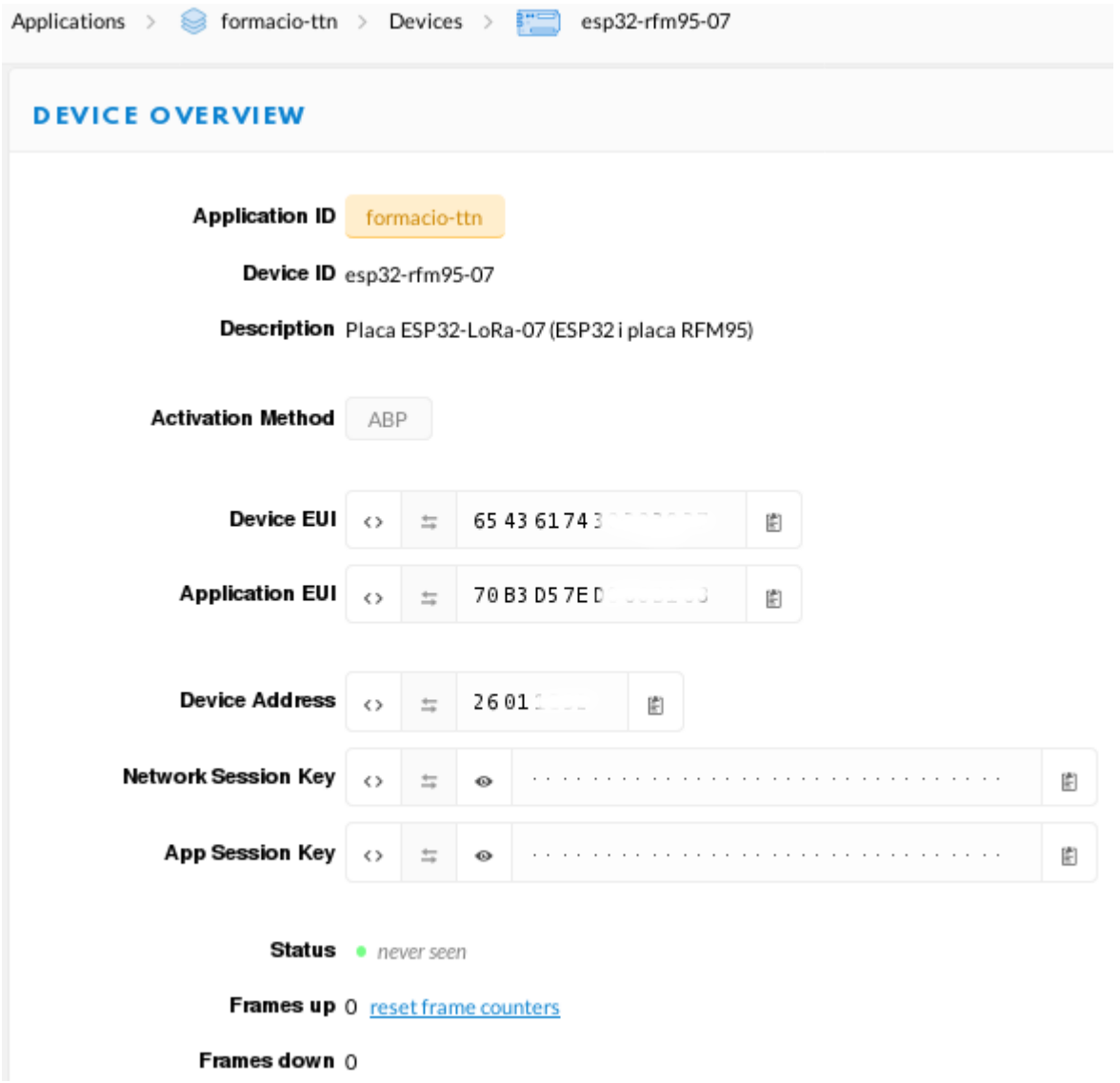
Registre de dispositiu a l'aplicació a TTN



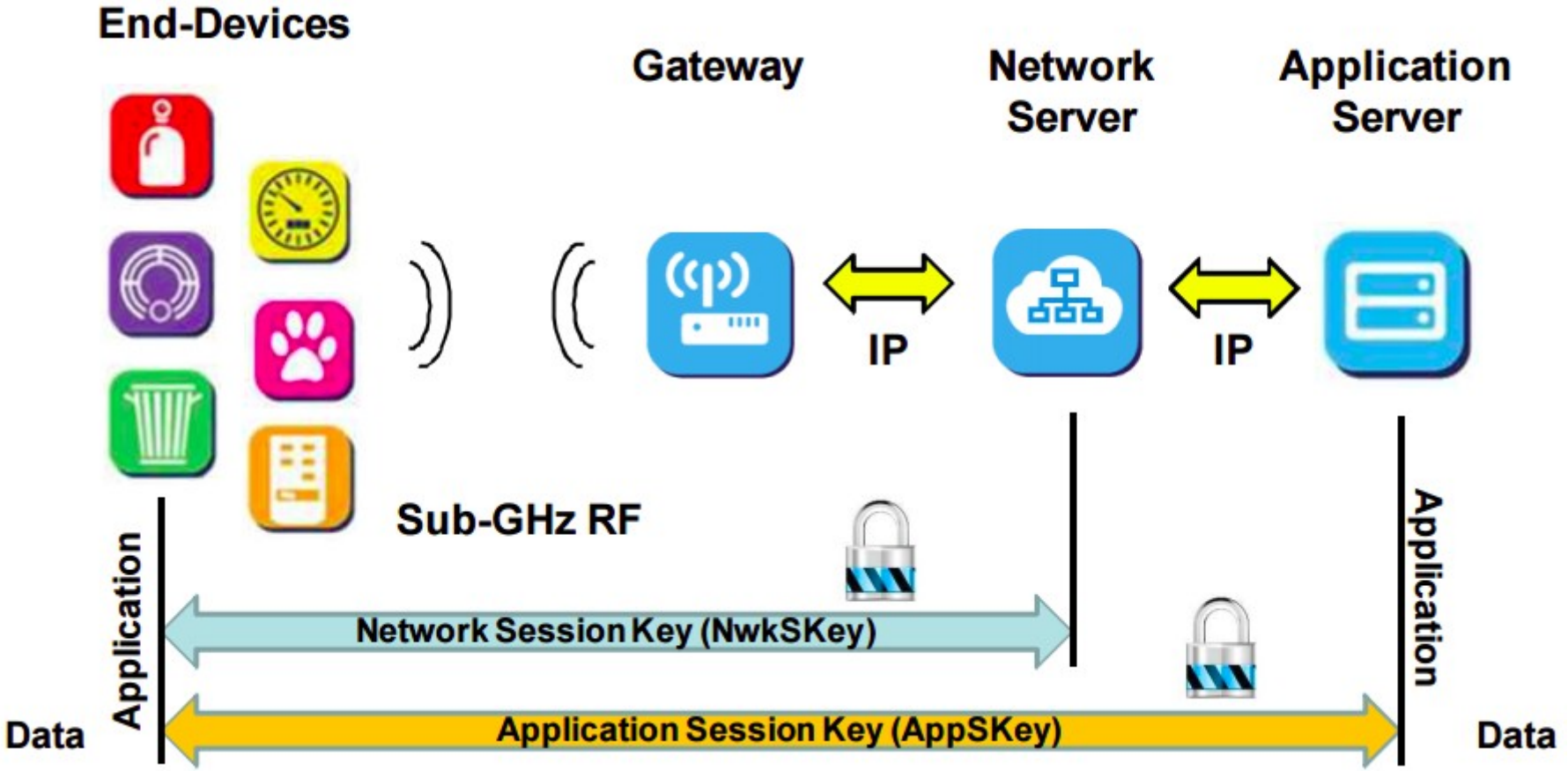
Registre de dispositiu a l'aplicació a TTN



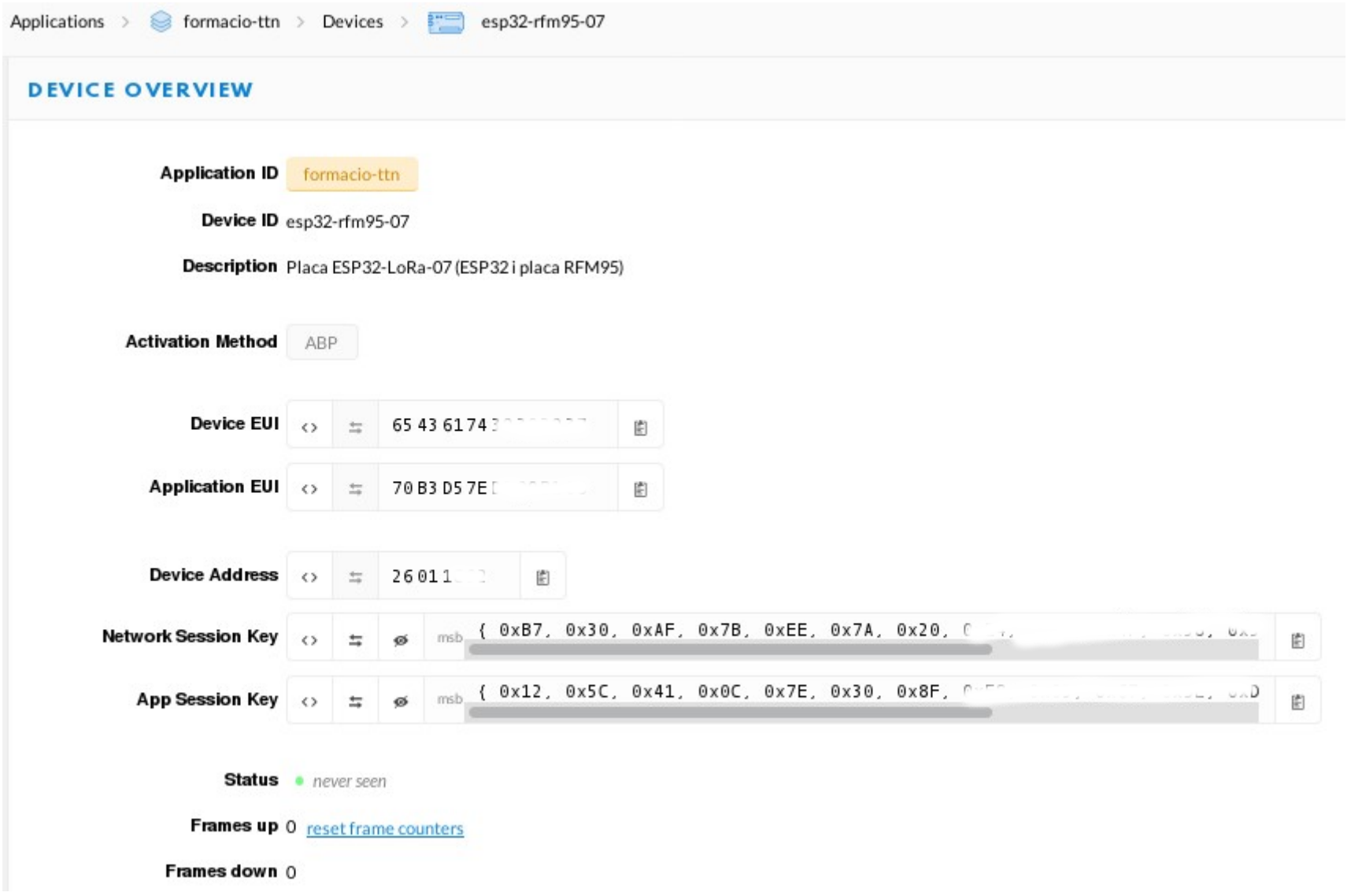
Registre de dispositiu a l'aplicació a TTN



LoRaWAN™



Registre de dispositiu a l'aplicació a TTN



Registre de dispositiu a l'aplicació a TTN

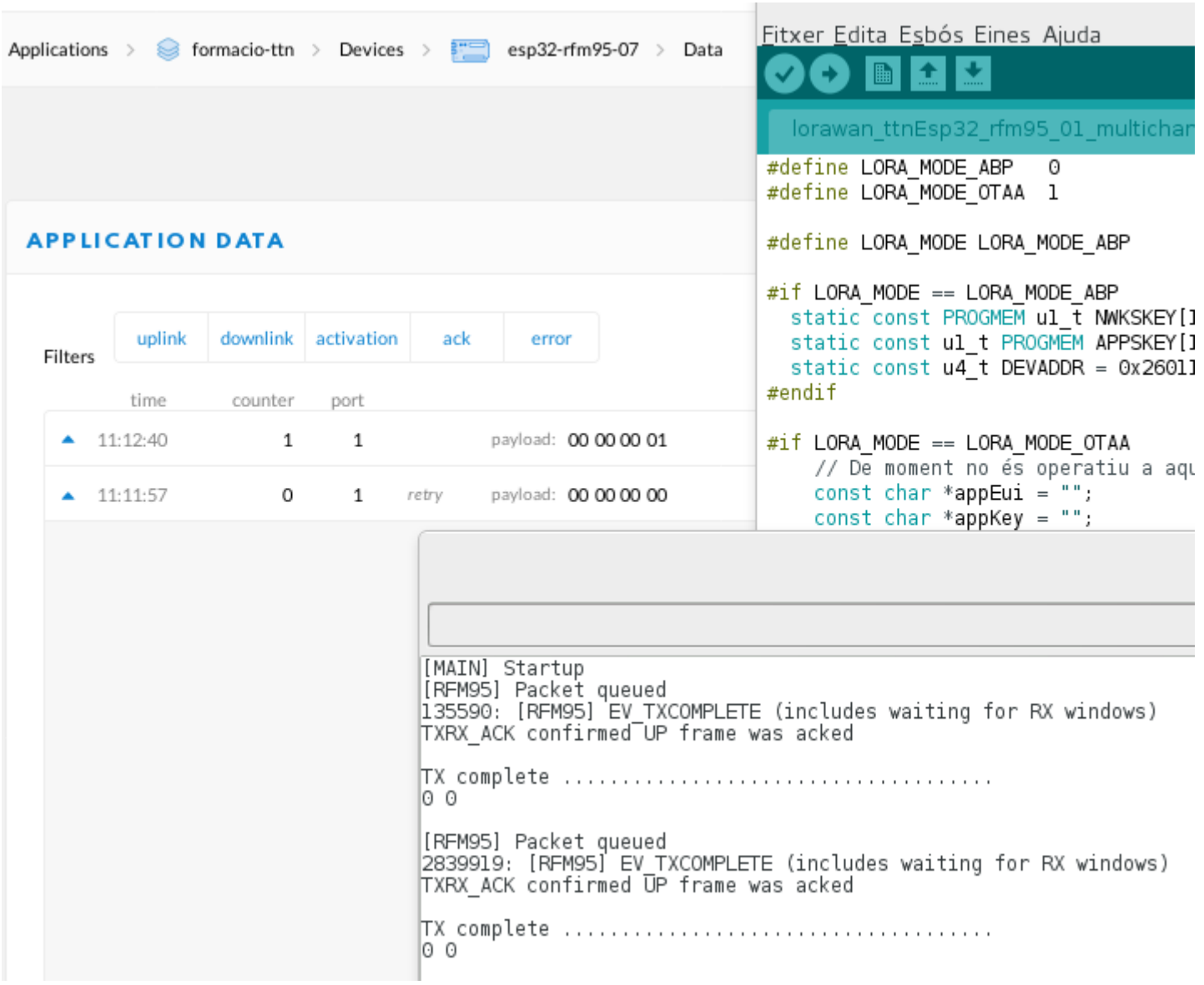
lorawan_ttnEsp32_rfm95_01_multichannel-esp32-rfm95-07 - credentials.h | Arduino 1.8.5

Fitxer Edita Esbós Eines Ajuda

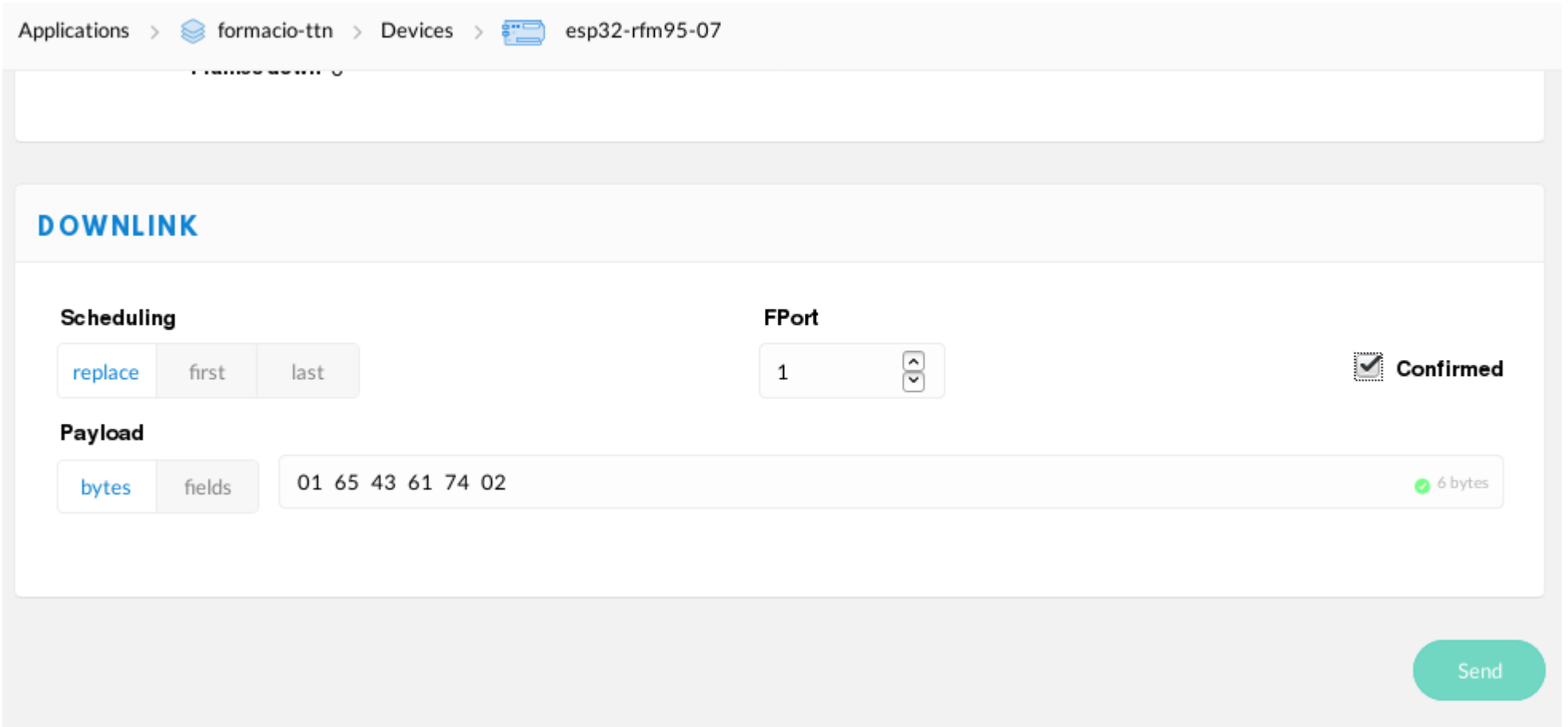
lorawan_ttnEsp32_rfm95_01_multichannel-esp32-rfm95-07 | credentials.h | pinMapping.h

```
#define LORA_MODE_ABP   0
#define LORA_MODE_OTAA  1

#define LORA_MODE LORA_MODE_ABP

#if LORA_MODE == LORA_MODE_ABP
  static const PROGMEM u1_t NWKSKEY[16] = { 0xB7, 0x30, 0xAF, 0x7B, 0xEE, 0x7A, 0x20, 0
  static const u1_t PROGMEM APPSKEY[16] = { 0x12, 0x5C, 0x41, 0x0C, 0x7E, 0x30, 0x8F,
  static const u4_t DEVADDR = 0x2601
#endif

#if LORA_MODE == LORA_MODE_OTAA
    // De moment no és operatiu a aquesta versió
    const char *appEui = "";
    const char *appKey = "";
#endif
```

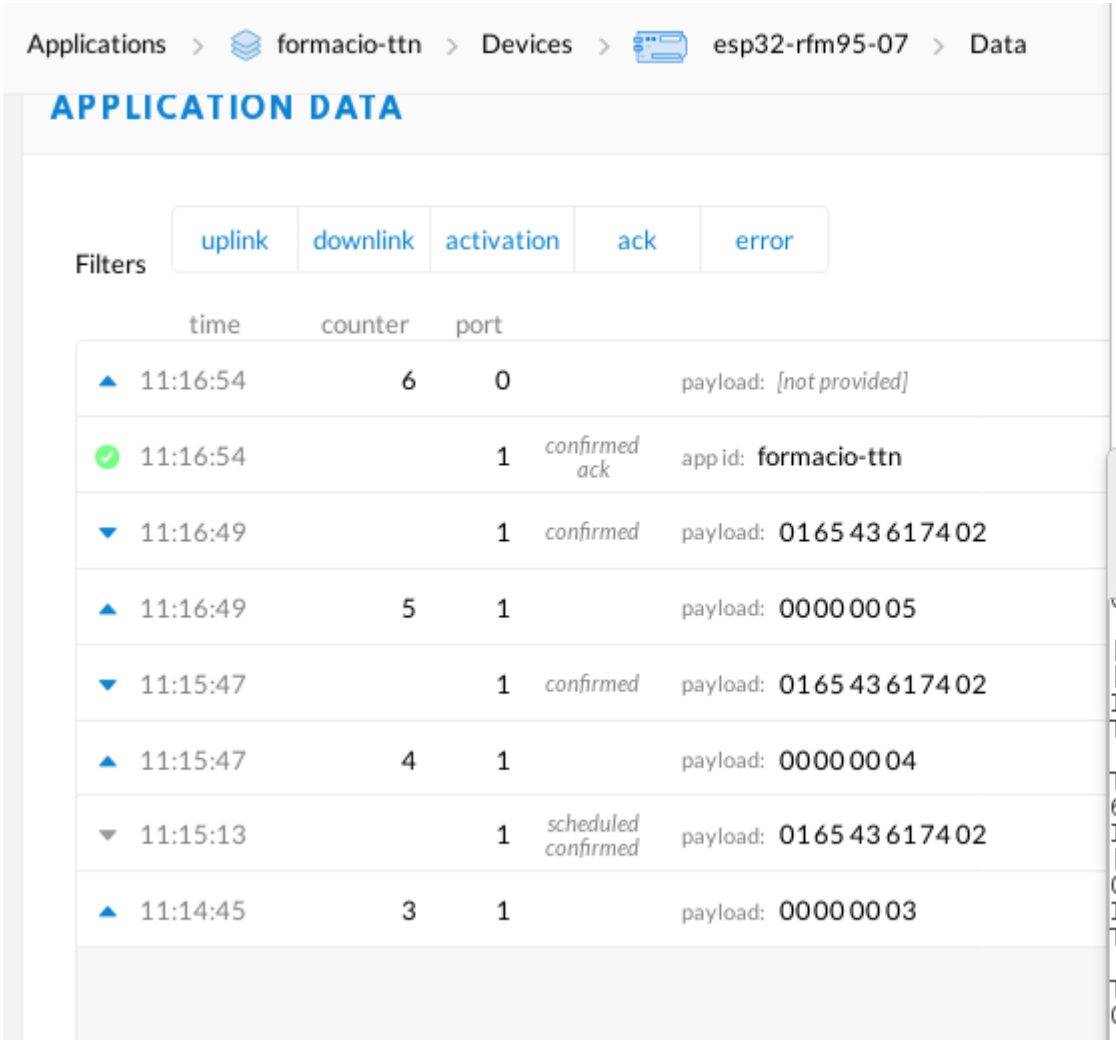
Uplink des del node a l'aplicació a TTN



Downlink des de l'aplicació TTN al node



Downlink des de l'aplicació TTN al node




```
#define LORA_MODE LORA_MODE_ABP

#if LORA_MODE == LORA_MODE_ABP
  static const PROGMEM u1_t NWKSKEY[
  static const u1_t PROGMEM APPSKEY[
  static const u4_t DEVADDR = 0x2601
#endif

#if LORA_MODE == LORA_MODE_OTAA
    // De moment no és operatiu a aq
    const char *appEui = "";
    const char *appKey = "";
#endif
```

```
[RFM95] Packet queued
[RFM95] Pending message
18295111: [RFM95] EV_TXCOMPLETE (incl
TXRX_ACK confirmed UP frame was acked

TX complete ......................
6 1
1654361742
[RFM95] Received 6 bytes of payload
01 65 43 61 74 02
18678004: [RFM95] EV_TXCOMPLETE (incl
TXRX_ACK confirmed UP frame was acked

TX complete ......................
0 0
```

Adaptació del node de multicanal a monocanal

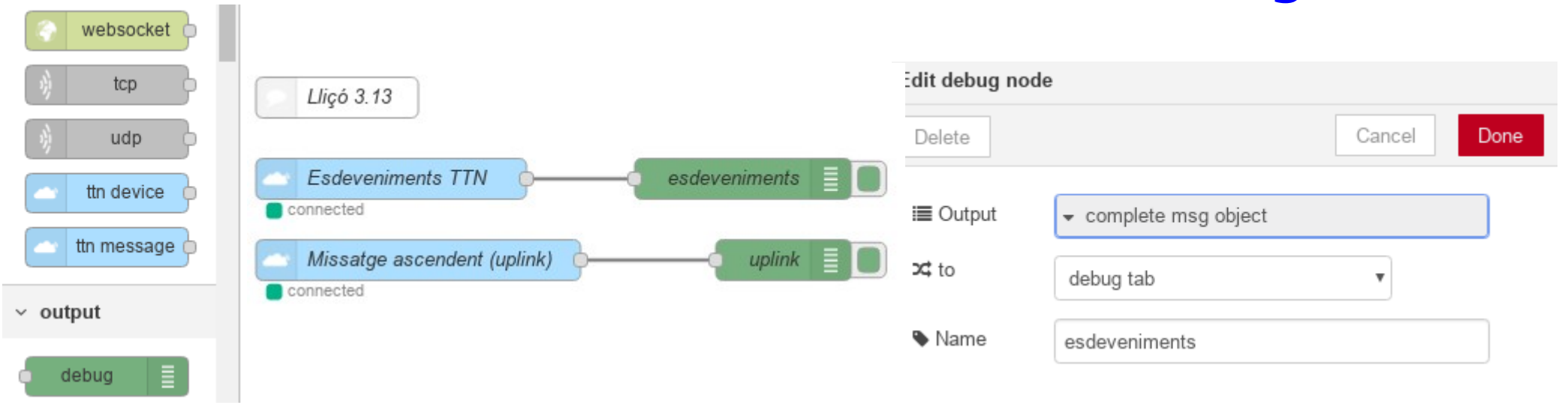
```
  LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
/*
  LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND_CENTI);      // g-band
  LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
  LMIC_setupChannel(3, 867100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
  LMIC_setupChannel(4, 867300000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
  LMIC_setupChannel(5, 867500000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
  LMIC_setupChannel(6, 867700000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
  LMIC_setupChannel(7, 867900000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
  LMIC_setupChannel(8, 868800000, DR_RANGE_MAP(DR_FSK,  DR_FSK),   BAND_MILLI);      // g2-band
*/
    LMIC_setupChannel(1, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
    LMIC_setupChannel(2, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
    LMIC_setupChannel(3, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
    LMIC_setupChannel(4, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
    LMIC_setupChannel(5, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
    LMIC_setupChannel(6, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
    LMIC_setupChannel(7, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);      // g-band
    LMIC_setupChannel(8, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7),   BAND_CENTI);
```

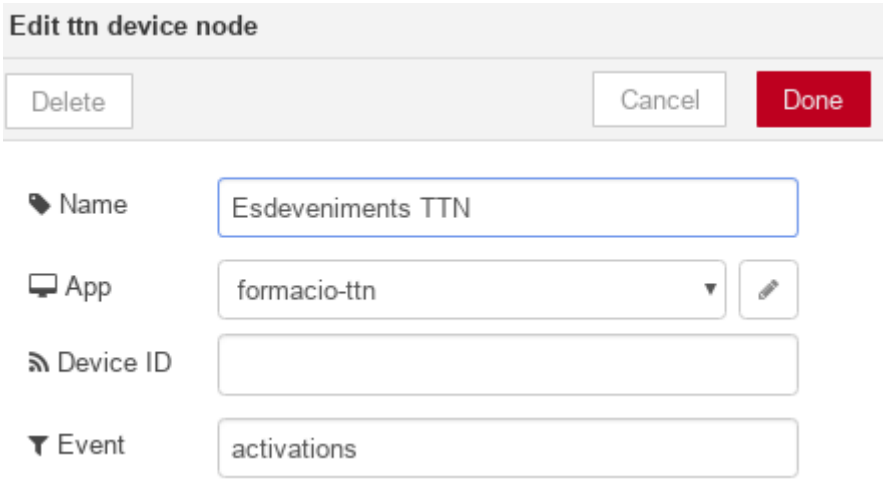
Enllaç al codi multicanal
Enllaç al codi monocanal
Biblioteca arduino-lmic

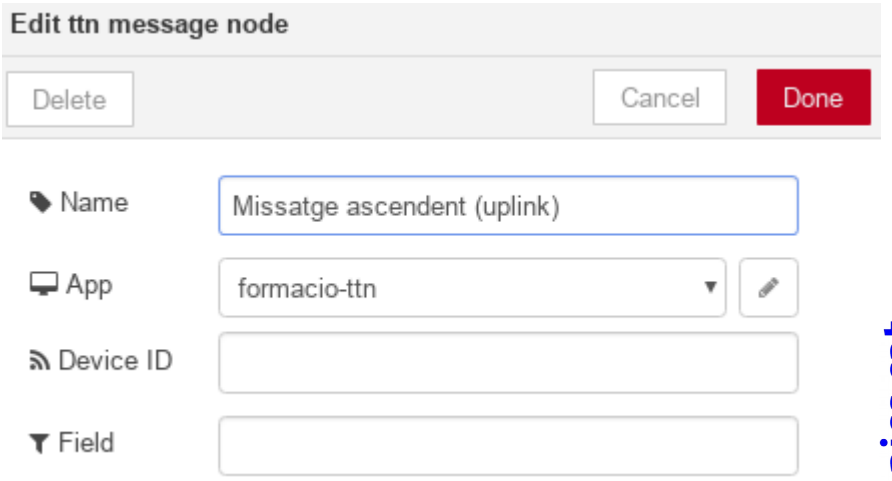
Node-RED

LoRaWAN

Nodes 'ttn device' i 'ttn message'





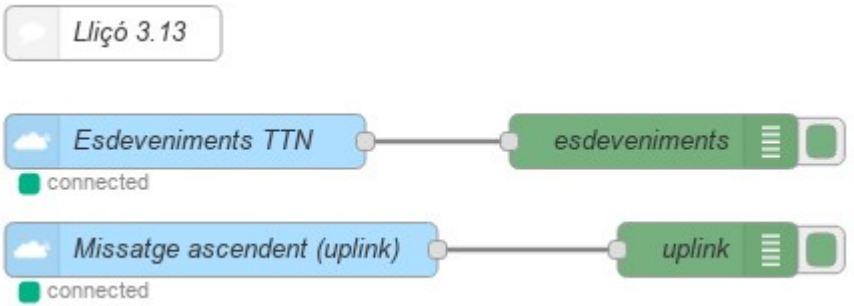


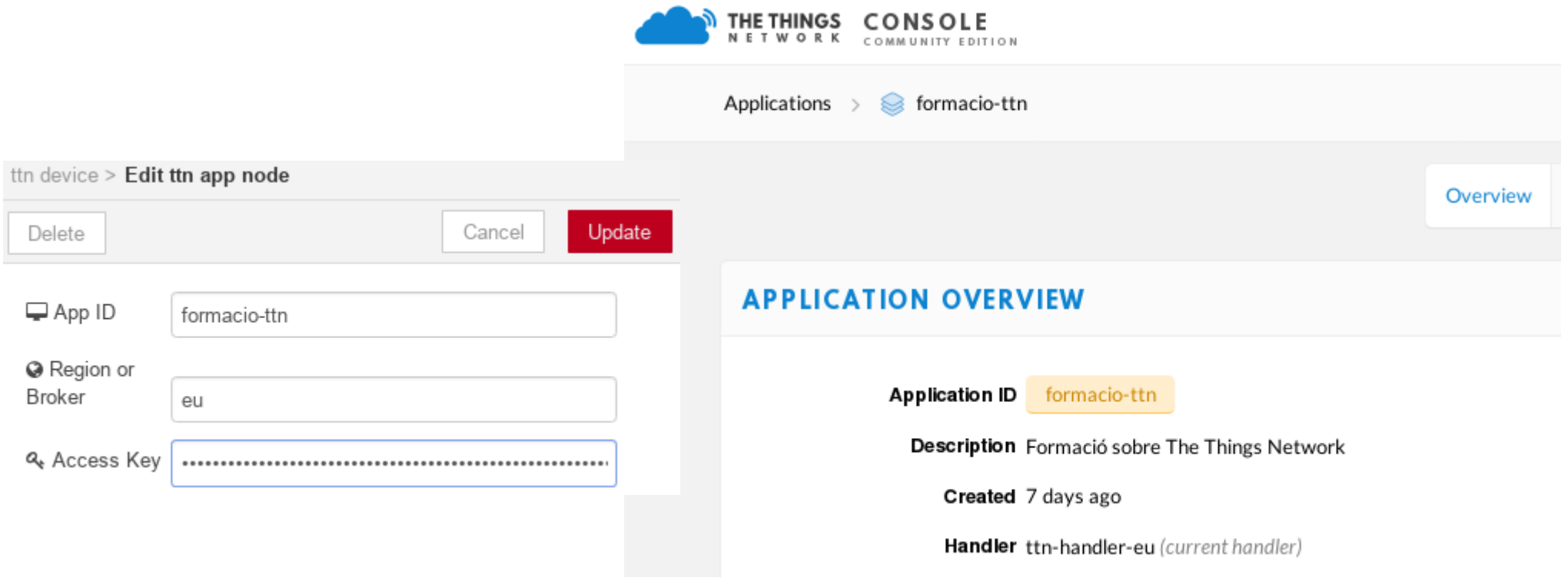
Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_13.txt
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

Node-RED

LoRaWAN

App ID



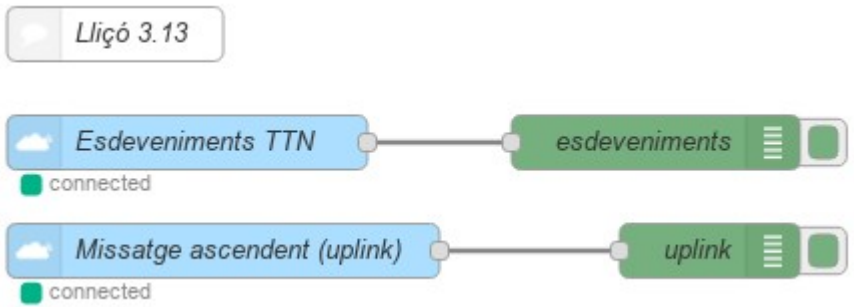


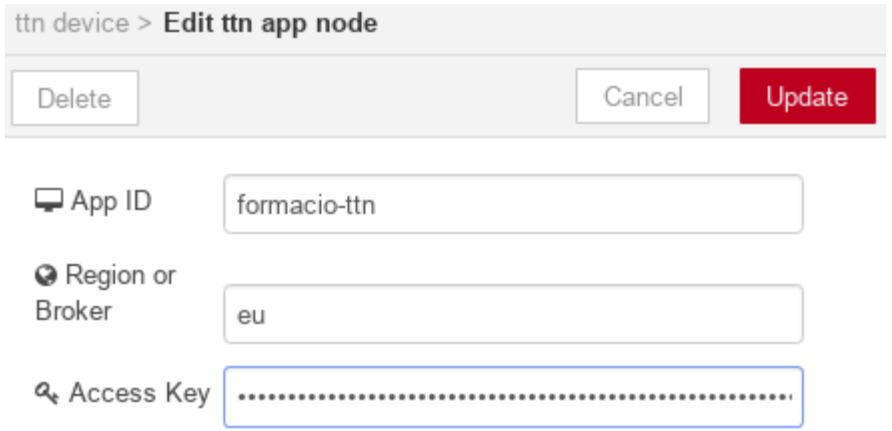
Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_13.txt
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

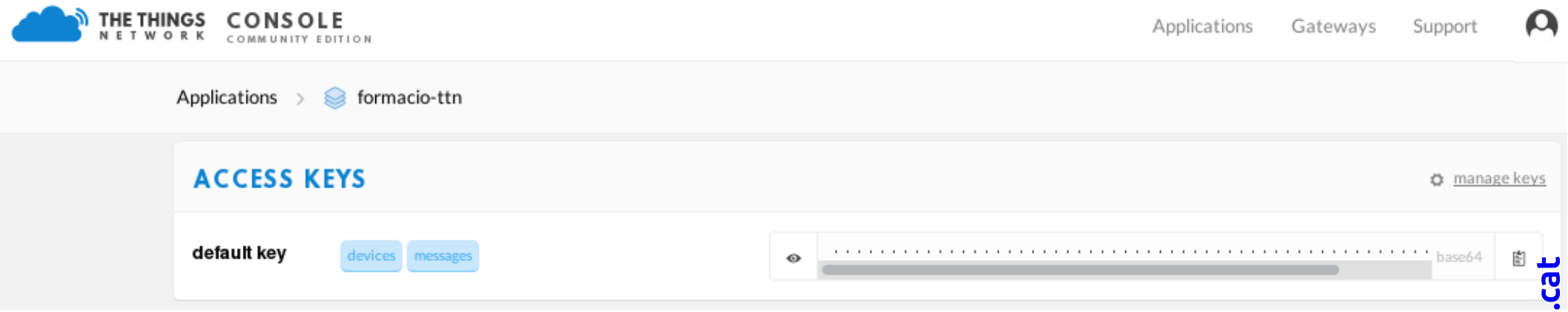
Node-RED

LoRaWAN

Access Key



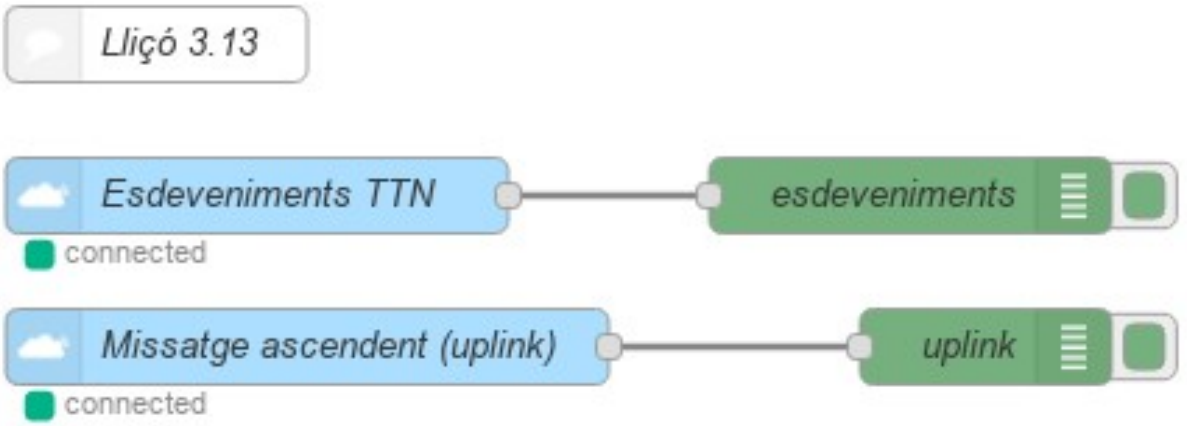


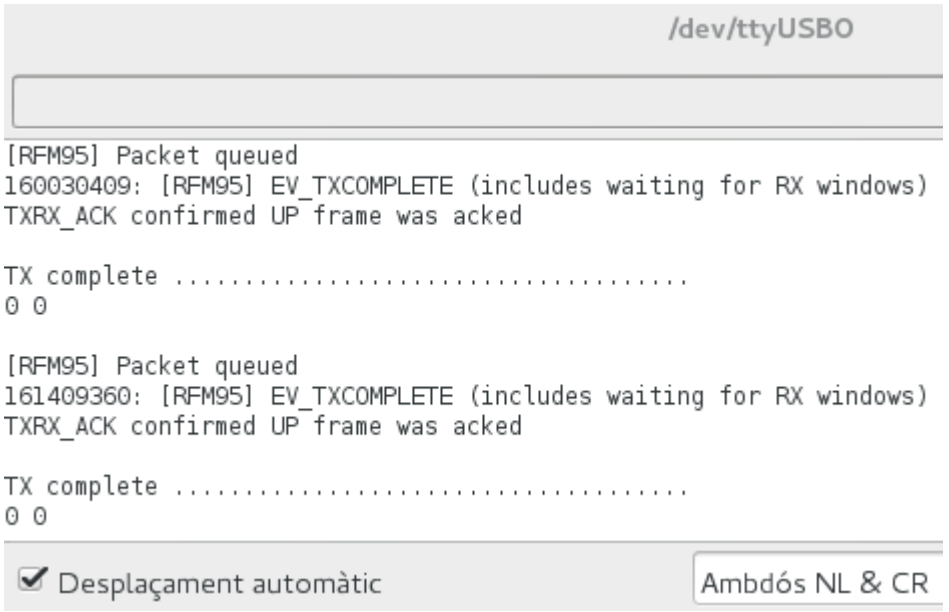


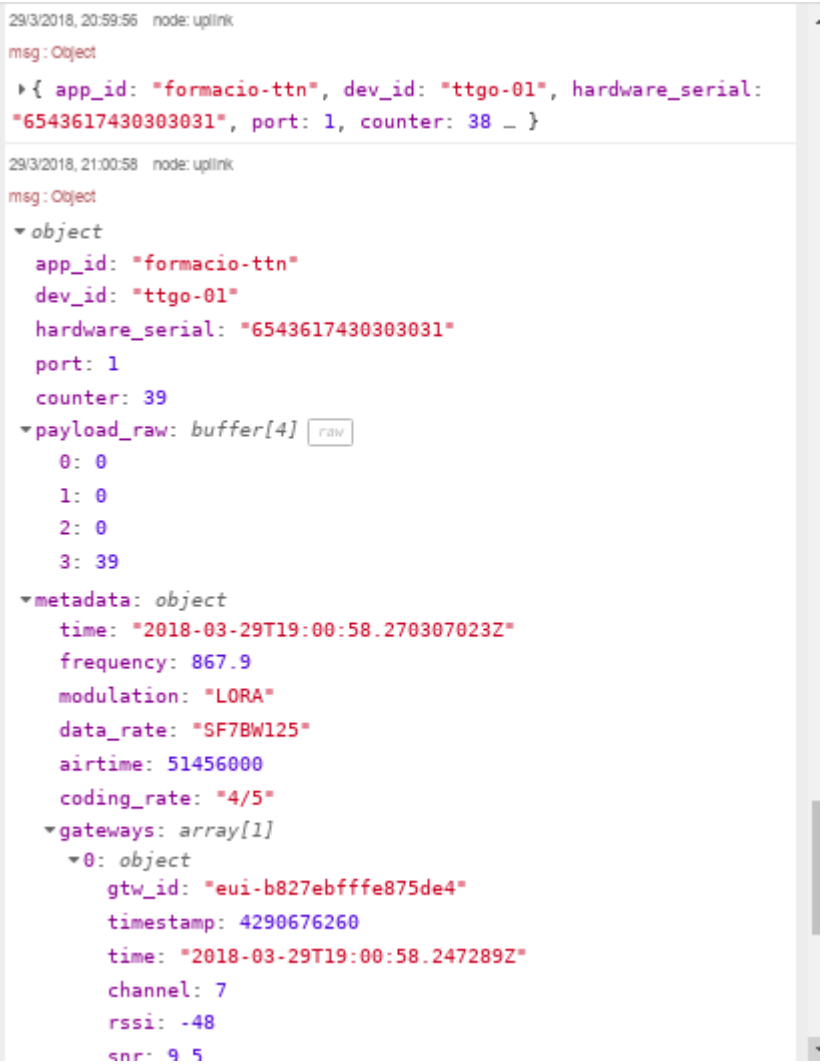
Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_13.txt
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

Node-RED

LoRaWAN





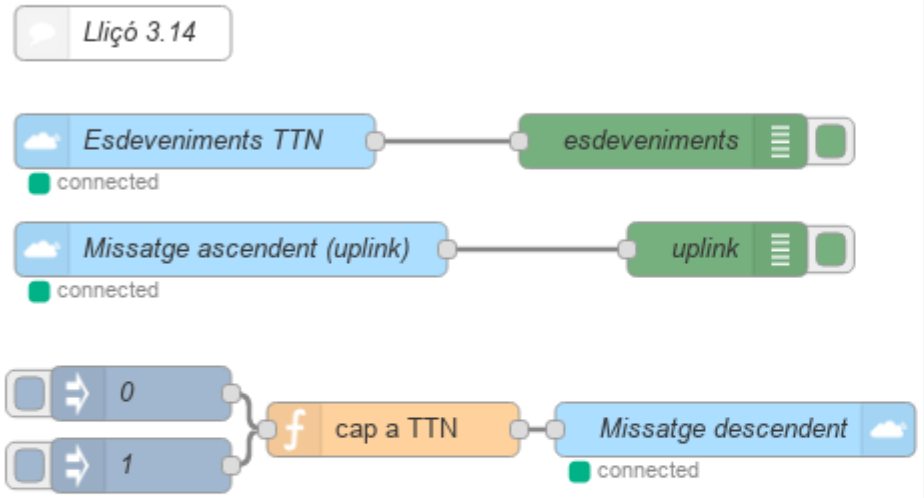


Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_13.txt
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

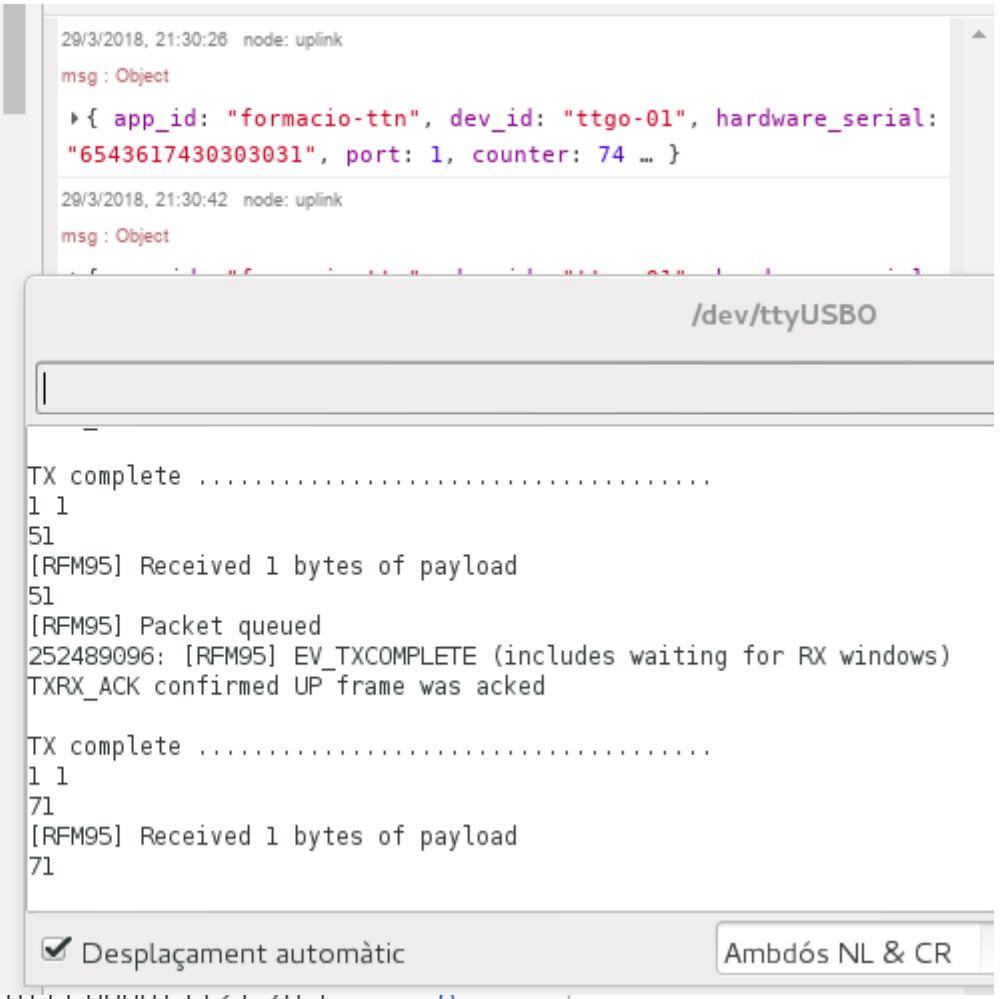
Node-RED

LoRaWAN

Missatge descendent (*downlink*)




```
return {
    dev_id: msg.dev_id,
    port: msg.port,
    schedule: "replace",
    confirmed: false,
    payload: Buffer.from([msg.payload ? 0x71 : 0x51])
}
```

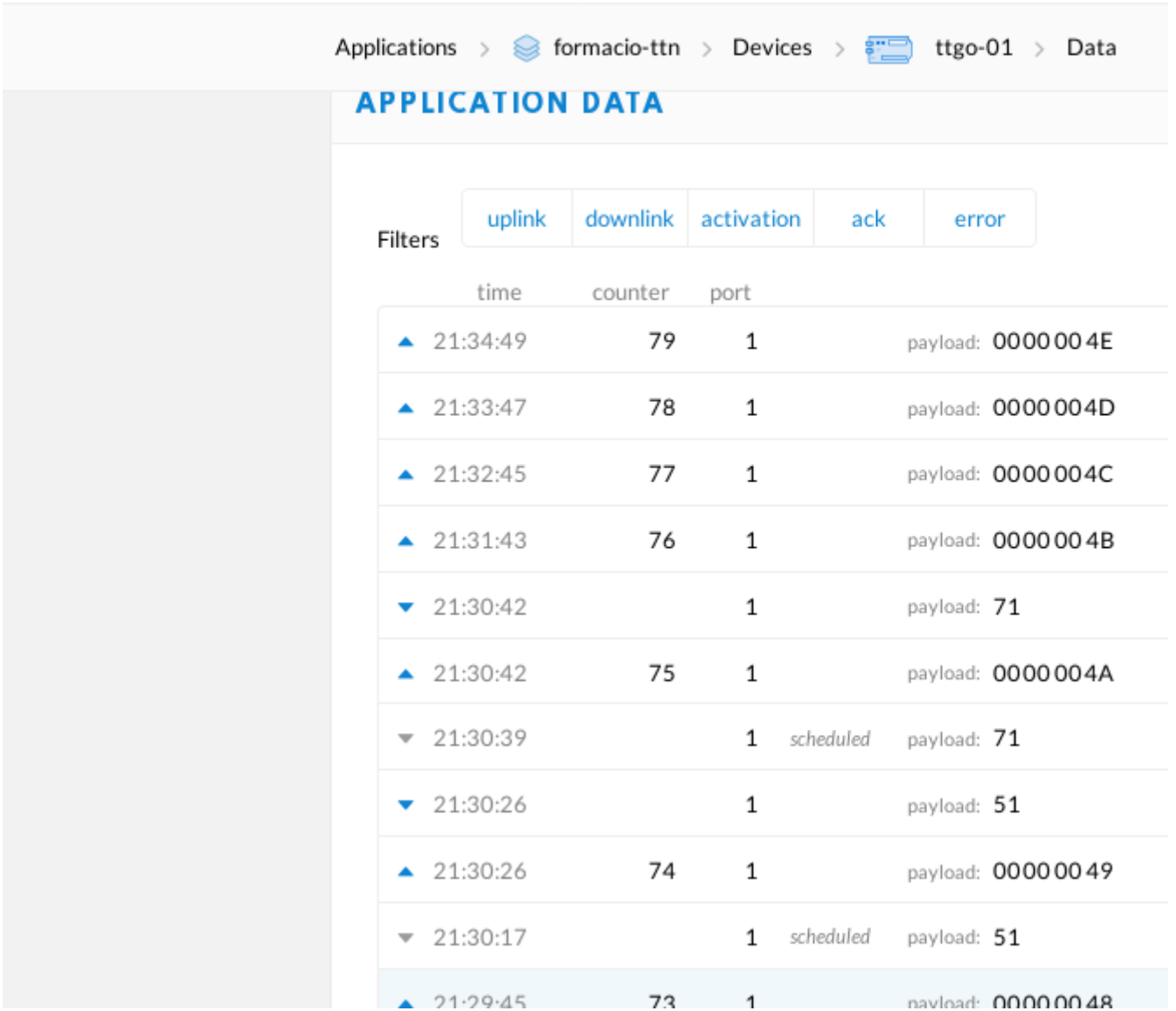


0101 0001	81	51	Q	0111 0001	113	71	q
0101 0010	82	52	R	0111 0010	114	72	r

Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_14.txt
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

Node-RED

LoRaWAN / Missatge descendent (*downlink*)



Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_14.txt
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

Node-RED

LoRaWAN

Missatge descendent (*downlink*)

Edit function node

Delete | Cancel | Done

Name: cap a TTN

Function

```
1  return {
2      dev_id: msg.dev_id,
3      port: msg.port,
4      schedule: "replace",
5      confirmed: false,
6      payload: Buffer.from([msg.payload ? 0x71 : 0x51])
7  }
```

Edit inject node

Delete | Cancel | Done

Payload: 0

Topic:

Repeat: none

Inject once at start?

Name: 0

0101 0000	80	50	P	0111 0000	112	70	p
0101 0001	81	51	Q	0111 0001	113	71	q
0101 0010	82	52	R	0111 0010	114	72	r

DEVICE OVERVIEW

Application ID: formacio-ttn

Device ID: ttgo-01

Description: Placa TTGO-01

Activation Method: ABP

Edit ttn send node

Delete | Cancel | Done

Name: Missatge descendent

App: formacio-ttn

Device ID: ttgo-01

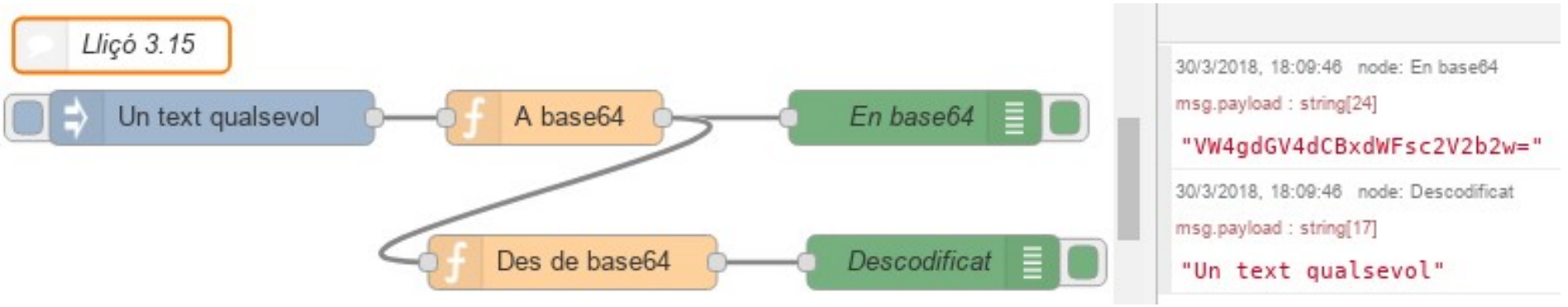
Port: 1

Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_14.txt

Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

Node-RED

Codificació i descodificació en base64

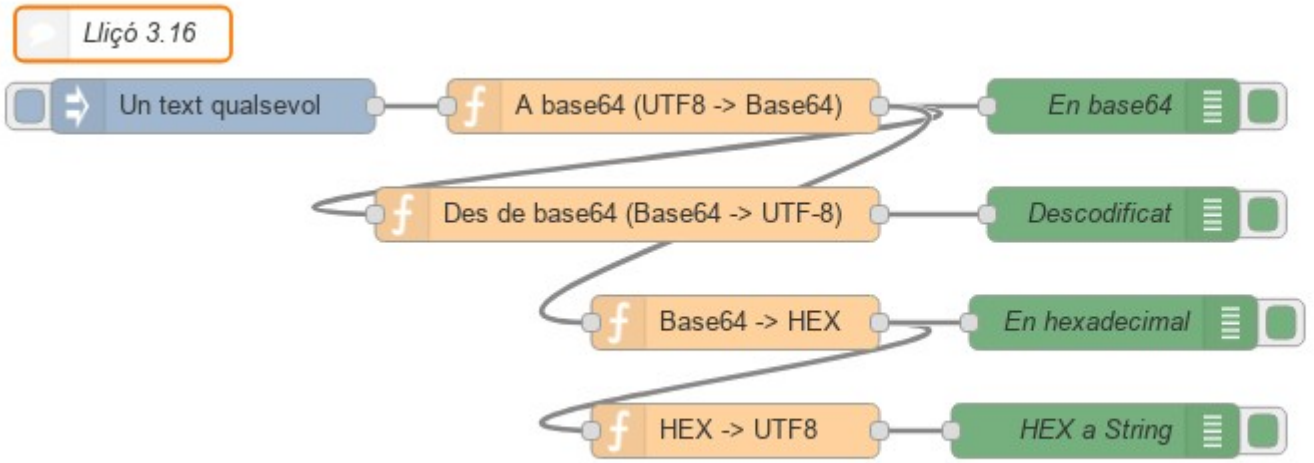


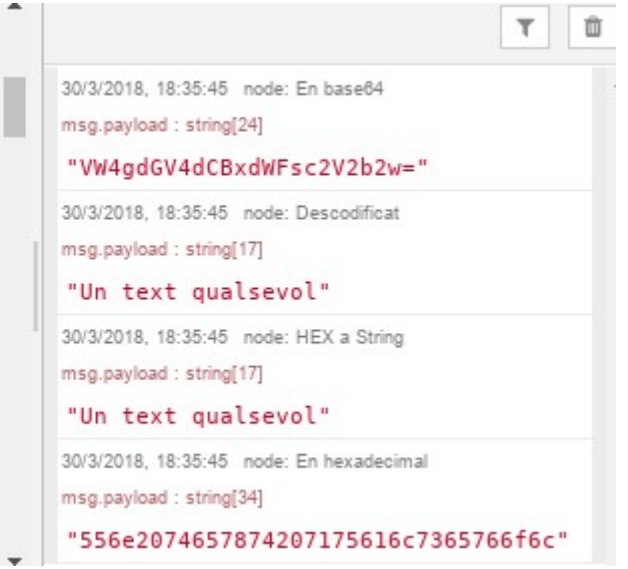
Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_15.txt
Per a codificar i descodificar en línia: https://www.base64encode.org/

Node-RED

Codificació i descodificació en base64

Presentació en hexadecimal



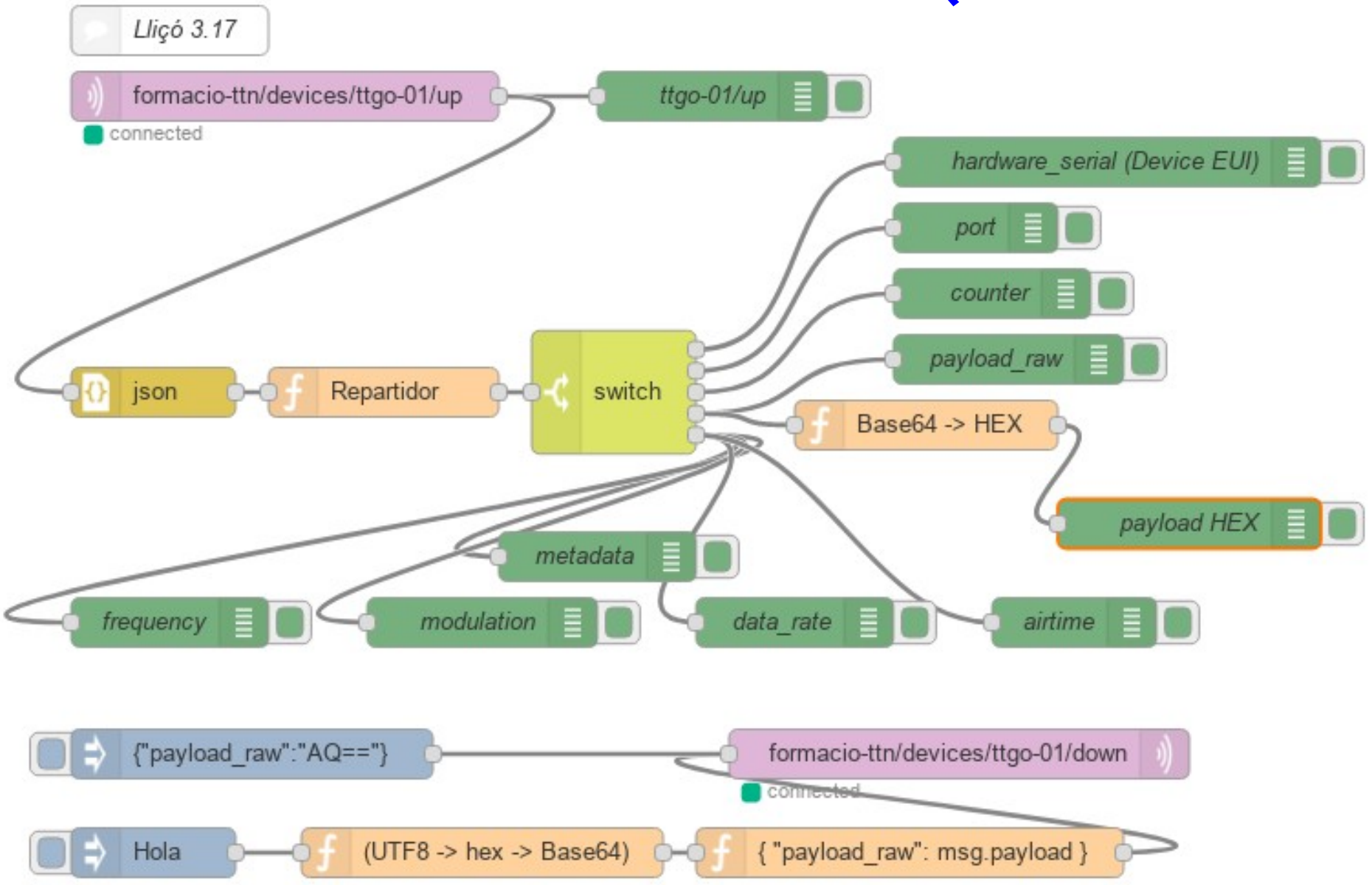


Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_16.txt

Node-RED

LoRaWAN

TTN fent servir MQTT



Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_17.txt

Node-RED

LoRaWAN

TTN fent servir MQTT

28

Edit mqtt in node

Delete | Cancel | Done

Server: eu.thethings.network:1883

Topic: formacio-ttn/devices/ttgo-01/up

QoS: 2

Name: Name

Edit mqtt out node

Delete | Cancel | Done

Server: eu.thethings.network:1883

Topic: formacio-ttn/devices/ttgo-01/down

QoS: | Retain:

Name: Name

Tip: Leave topic, qos or retain blank if you want to set them via msg properties.

mqtt in > **Edit mqtt-broker node**

Delete | Cancel | Update

Connection | Security | Birth Message | Will Message

Server: eu.thethings.network Port: 1883

Enable secure (SSL/TLS) connection

Client ID: Leave blank for auto generated

Keep alive time (s): 60 Use clean session

Use legacy MQTT 3.1 support

mqtt in > **Edit mqtt-broker node**

Delete | Cancel | Update

Connection | Security | Birth Message | Will Message

Username: formacio-ttn

Password: ••••••••

Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_17.txt

Node-RED

LoRaWAN

TTN fent servir MQTT

Edit function node

Delete | Cancel | Done

Name: Repartidor

Function

```
1  var keys = Object.keys(msg.payload);
2
3  var msgs = keys.map(function(key) {
4      return { topic: key, payload: msg.payload[key] };
5  });
6  return [msgs];
```

Edit switch node

Delete | Cancel | Done

Name: Name

Property: msg. topic

==	hardware_serial	→ 1
==	port	→ 2
==	counter	→ 3
==	payload_raw	→ 4
==	metadata	→ 5

Edit function node

Delete | Cancel | Done

Name: (UTF8 -> hex -> Base64)

Function

```
1  var b = new Buffer(msg.payload);
2  msg.payload = b.toString('base64');
3
4  return msg;
```

Edit function node

Delete | Cancel | Done

Name: { "payload_raw": msg.payload }

Function

```
1  msg.payload = { "payload_raw": msg.payload };
2
3  return msg;
```

Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_17.txt