

LoRaWAN Gateways

Angel Luis Martínez (aka AngeLinux)
Twitter: @angelinux99

¿Qué es LoRa?

LoRa es un tipo de modulación de radiofrecuencia patentado por Semtech con las siguientes funcionalidades:

- Alta tolerancia a las interferencias
- Alta sensibilidad para recibir datos (-168dB)
- Basado en modulación CHIRP
- Bajo consumo (hasta 10 años con una batería*)
- Largo alcance 10 a 20km
- Baja transferencia de datos (hasta 255 bytes)
- Conexión punto a punto
- Frecuencias de trabajo: 915Mhz América, 868 Europa, 433 Asia.

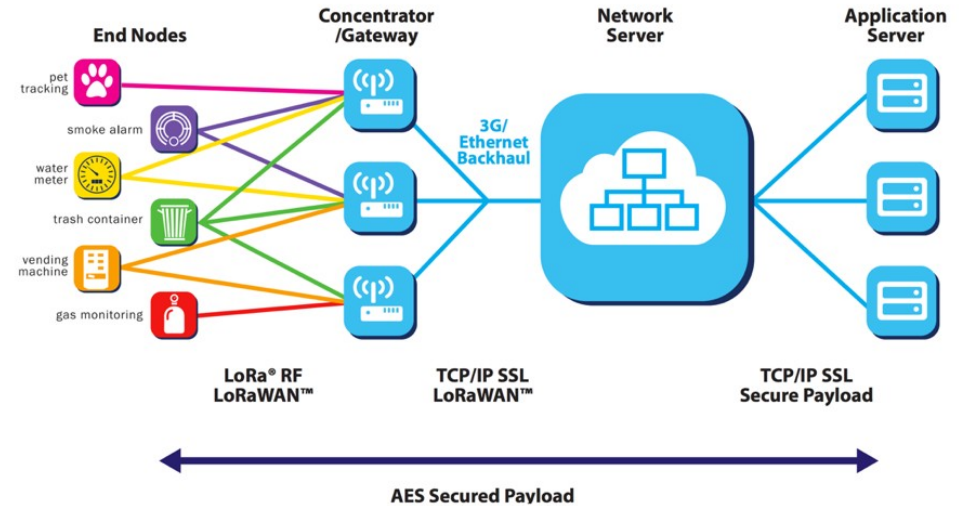


¿Qué es LoRaWAN?

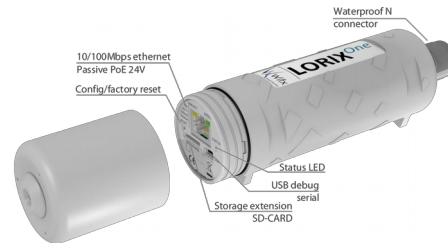
LoRaWAN define el protocolo de comunicación y la arquitectura del sistema para la red. Administra las frecuencias de comunicación, la velocidad de datos y la potencia de todos los dispositivos. Los dispositivos en la red son asíncronos y transmiten cuando tienen datos disponibles para enviar. Los datos transmitidos por un dispositivo de nodo final son recibidos por múltiples puertas de enlace, que envían los paquetes de datos a un servidor de red centralizado. El servidor de red filtra paquetes duplicados, realiza comprobaciones de seguridad y administra la red y los datos se envían a los servidores de aplicaciones.

Características LoRa y el protocolo LoRaWAN

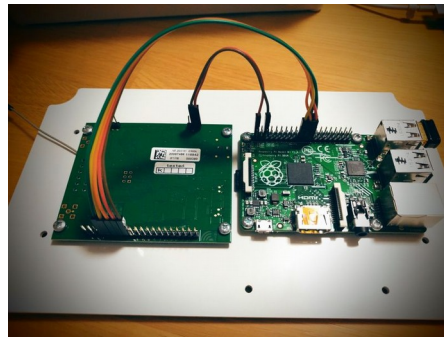
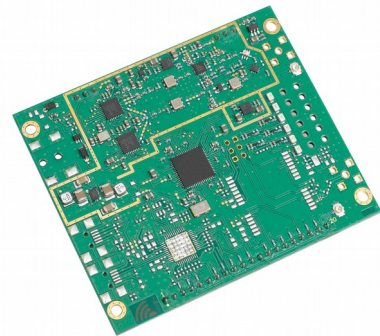
- Red de bajo coste
- Cumple con standards
- Nodos de bajo consumo
- Red de amplio rango
- Seguridad end-to-end (AES128)
- Alta capacidad (millones de mensajes por gateway)



Concentradores / Gateways



Concentradores / Gateways (DIY)



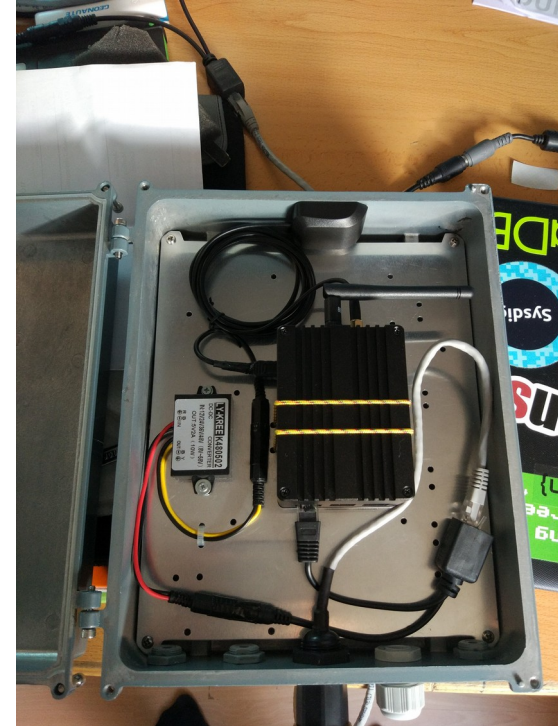
Recomendaciones (1)

- Instalación con PoE pasivo, y comunicación por ethernet. Como alimentación, podemos utilizar un viejo alimentador de ordenador portátil.
- Ubicación de GW en exterior con una buena antena externa



Recomendaciones (2)

- Caja externa estanca.
Prensaestopas para cable ethernet
y para pigtail de antena.
- Pigtail de SMA a type N
- Instalación con PoE pasivo, y
comunicación por ethernet



Instalando el sistema operativo

-Descargar Raspbian Stretch Lite

<https://www.raspberrypi.org/downloads/raspbian/>

-Instalar Raspbian en MicroSD (Linux / MAC):

```
dd if=2018-11-13-raspbian-stretch-lite.img of=/dev/sdx bs=4M conv=fsync
```

-Instalar Raspbian en MicroSD (Windows): Descargar rufus e instalar

https://rufus.ie/es_ES.html

Una vez terminado el proceso, añadir un fichero vacío con el nombre "ssh" en la partición boot, para poder acceder desde ssh en el arranque.

Primer arranque (1)

- Insertar tarjeta MicroSD, conectar por ethernet al switch y conectar alimentación. Como no sabemos la IP, habrá que mirar en el servidor DHCP la IP que nos ha ofrecido. Una vez conocida la IP, podremos acceder por ssh (Linux/MAC) o Putty (Cliente SSH Windows).

```
pi@rak831:~ $ ssh pi@192.168.20.171
```

Primer arranque (2)

- Con el comando `ifconfig`, apuntar las MAC ADDRESS de la ethernet y de la WIFI para futuras referencias, las MACs son del tipo: MAC Address: `b8:27:eb:xx:xx:xx` (Raspberry Pi Foundation)

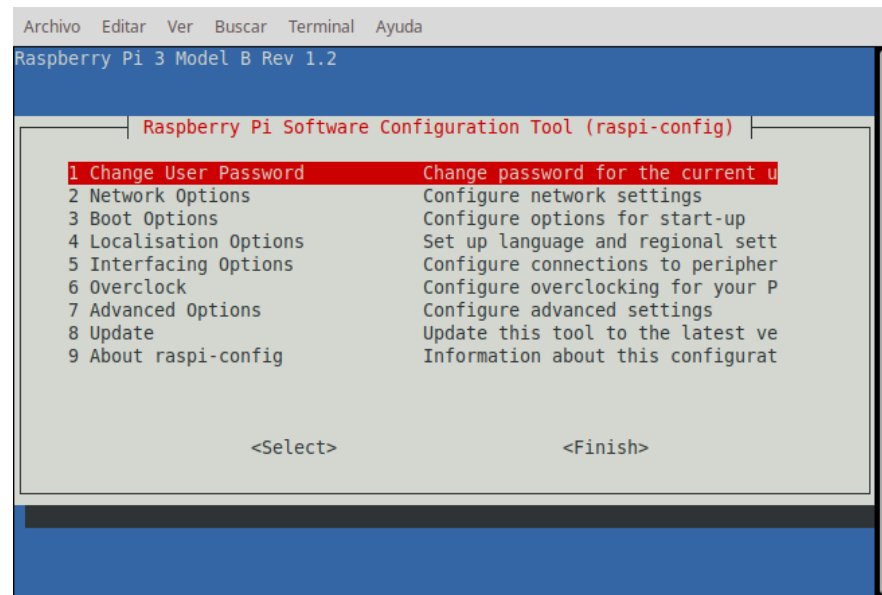
```
pi@rak831:~ $ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.20.180 netmask 255.255.255.0 broadcast 192.168.20.255
    ether b8:27:eb:xx:xx:xx txqueuelen 1000 (Ethernet)

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.20.171 netmask 255.255.255.0 broadcast 192.168.20.255
    inet6 fe80::70eb:c0ed:d07e:214f prefixlen 64 scopeid 0x20<link>
    ether b8:27:eb:88:00:ab txqueuelen 1000 (Ethernet)
```

raspi-config, la herramienta !!! (1)

- Vamos a configurar algunas cosas necesarias para el funcionamiento de nuestro gateway, con el siguiente comando :

```
pi@rak831:~ $ sudo raspi-config
```



raspi-config, la herramienta !!! (2)

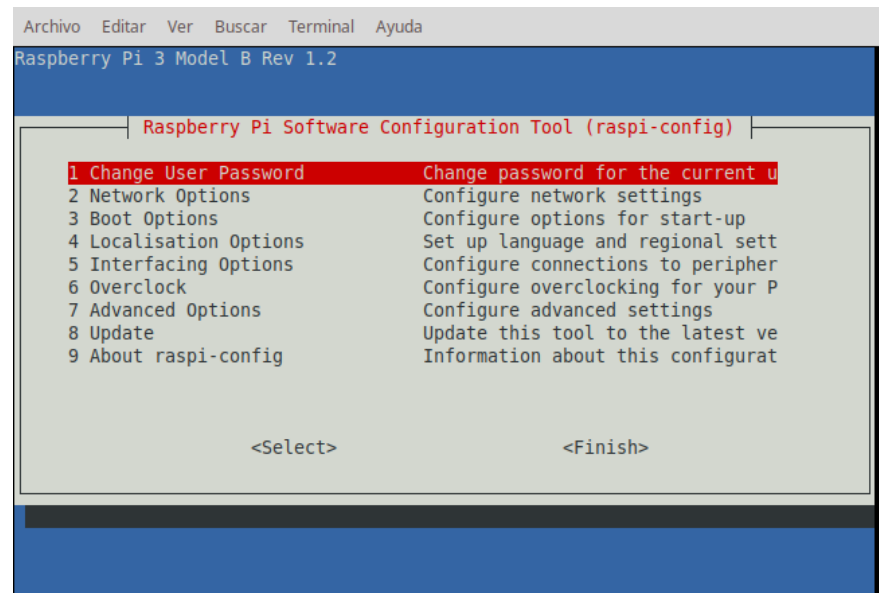
Cambiamos las siguientes opciones :

Change User Password (1)

Network Options (2)

N1 Hostname

N2 Wi-Fi (valores conexión wifi)



raspi-config, la herramienta !!! (3)

5 Interfacing Options

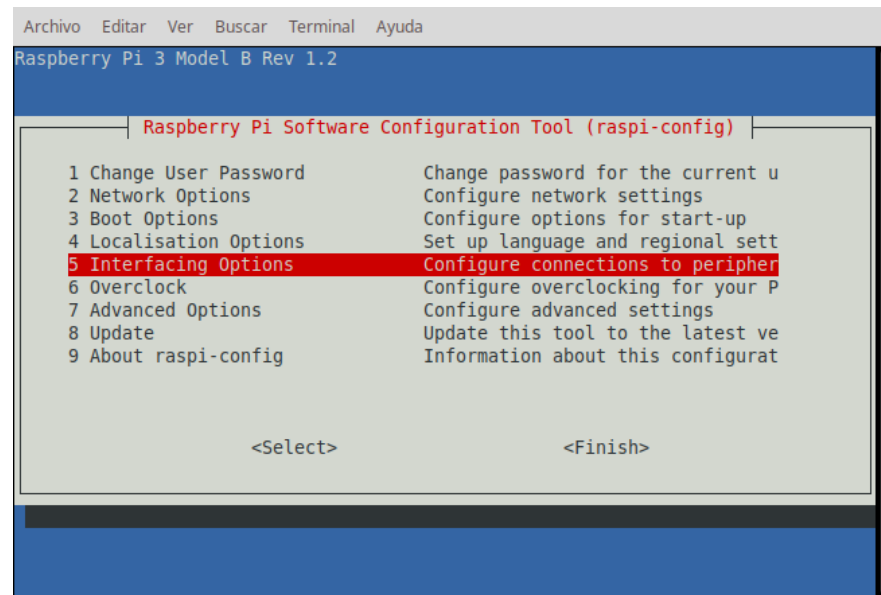
P4 SPI --> Habilitamos SPI

P6 Serial

--> Would you like a login shell to be accessible over serial? --> NO

--> Would you like the serial port hardware to be enabled? --> YES

Le damos a <Finish> y reiniciamos...



Actualización de paquetes

- Con la conexión a internet activa, actualizamos paquetes e instalamos alguno nuevos paquetes nuevos necesarios :

```
pi@rak831:~ $ sudo apt update
pi@rak831:~ $ sudo apt dist-upgrade
pi@rak831:~ $ sudo apt install git minicom
```

Comprobación de GPS

- Comprobamos GPS indicando puerto serie habilitado en la raspi :

```
pi@rak831:~ $ sudo minicom -c on -b 9600 -o -D /dev/ttyS0
```

- Deberían aparecer caracteres recibidos desde el GPS. Para salir de minicom, CTRL-A X

Semtech UDP Packet Forwarder (1)

- Clonamos el instalador y arrancamos la instalación :

```
$ git clone -b spi https://github.com/ttn-zh/ic880a-gateway.git
$ cd ~/ic880a-gateway
$ sudo ./install.sh spi
    - Enable Remote Configuration ? NO
```

Semtech UDP Packet Forwarder (2)

- Modificamos en el fichero de arranque el pin de RESET (para rak831, pin 17. Para ic880a, pin 25):

```
$ sudo vim /opt/ttn-gateway/bin/start.sh
```

```
# Reset iC880a PIN  
SX1301_RESET_BCM_PIN=17
```

Semtech UDP Packet Forwarder (3)

- Modificamos el fichero de configuración, mantenemos todo igual y modificar / añadir las siguientes líneas. Añadir valores reales de posición definitiva del gw en ref_latitude, ref_longitude y ref_altitude :

```
$ sudo vim /opt/ttn-gateway/bin/local_conf.json
```

```
/* GPS configuration */  
"gps_tty_path": "/dev/ttyS0",  
"fake_gps": false,  
"ref_latitude": 10,  
"ref_longitude": 20,  
"ref_altitude": 670,
```

Registro de gateway en TTN (1)

- Para configurar el gateway necesitamos añadirlo a la consola TTN :
 - Utiliza tu usuario TTN para loguearte en la siguiente URL:
`https://console.thethingsnetwork.org/`
 - En la pantalla principal, hacer click en “gateways” y a continuación “register gateway”
 - Seleccionar “legacy packet forwarder” (Semtech packet fw)
 - Introducir el Gateway EUI (el que hemos apuntado anteriormente en la instalación del packet forwarder), y añadir una descripción del gateway

Registro de gateway en TTN (2)

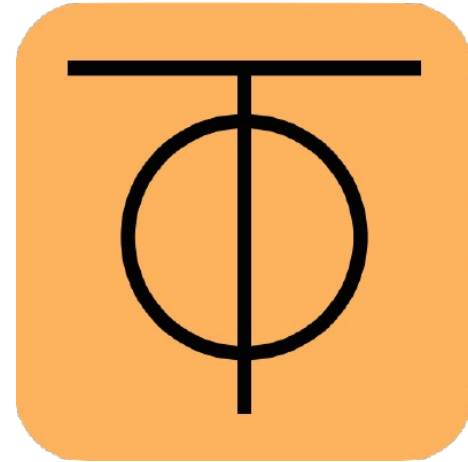
- Frequency Plan: "Europe 868Mhz"
- Router: `ttn-router-eu`
- Location: Hacer zoom en el mapa y ubicar la posición final del gateway. En "lat y lng" aparecerá el punto donde vas a ubicar el GW. En TTN éstos datos no se actualizan automáticamente.
- Antenna Placement: Indicar si es indoor / outdoor
- Por último, seleccionar el botón "Register Gateway"
- Una vez registrado, podremos acceder a "Settings" en la parte superior derecha, para indicar la altitud, la privacidad de los datos información de la marca y modelo y otras opciones.

Algunas “chuches” (1)

- ZeroTier : Directly Connecting the World's Devices with Universal Software Defined Networking. Con éste software puedes gestionar hasta 100 dispositivos dentro de una misma red o en distintas redes.

<https://www.zerotier.com/>

<https://github.com/zerotier>



Algunas “chuches” (2)

- Mostrar la temperatura de la CPU y GPU de nuestra RaspberryPi

```
#!/bin/bash
# Script: my-pi-temp.sh
# Purpose: Display the ARM CPU and GPU temperature of Raspberry Pi
2/3
# Author: Vivek Gite <www.cyberciti.biz> under GPL v2.x+
# -----
cpu=$(
```

`</sys/class/thermal/thermal_zone0/temp)`

```
echo "$(date) @ $(hostname) "
echo "-----"
echo "GPU => $(/opt/vc/bin/vcgencmd measure_temp) "
echo "CPU => $((cpu/1000)) 'C"
```

Algunas “chuches” (2)

- Mostrar la temperatura de la CPU y GPU de nuestra RaspberryPi

```
#!/bin/bash
# Script: my-pi-temp.sh
# Purpose: Display the ARM CPU and GPU temperature of Raspberry Pi
2/3
# Author: Vivek Gite <www.cyberciti.biz> under GPL v2.x+
# -----
cpu=$(
```

`</sys/class/thermal/thermal_zone0/temp)`

```
echo "$(date) @ $(hostname) "
echo "-----"
echo "GPU => $(/opt/vc/bin/vcgencmd measure_temp) "
echo "CPU => $((cpu/1000)) 'C"
```