ED-GWL2110

User Manual

by EDA Technology Co., Ltd

built: 2025-08-01

1 Product Overview

The ED-GWL2110 is an outdoor gateway based on the Raspberry Pi Compute Module 4 (CM4). The device features an all-aluminum alloy sealed enclosure, providing excellent weatherproof, waterproof, insect-resistant, and lightning-resistant performance. It supports multiple LoRa modules across various frequency bands (requires external antennas of corresponding frequencies). Optional 4G and Wi-Fi/BT connectivity ensure reliable data upload and download in outdoor environments.

Equipped with an onboard GNSS module, the ED-GWL2110 enables precise positioning capabilities. Its built-in hardware watchdog module, Real-Time Clock (RTC), and encryption chip further enhance system security and reliability.

1.1 Target Application

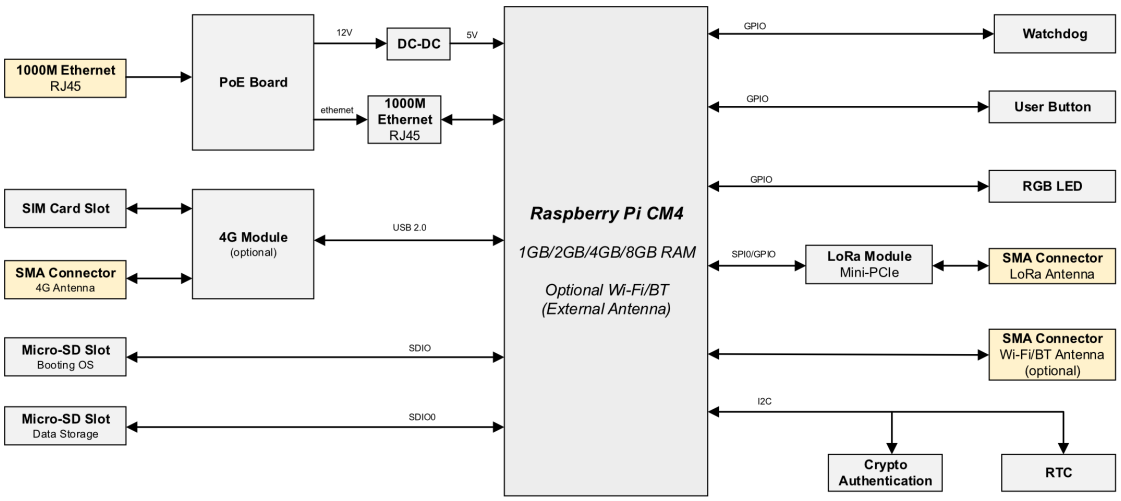
- Smart City
- Smart Transportation
- Smart Agriculture

1.2 Specifications

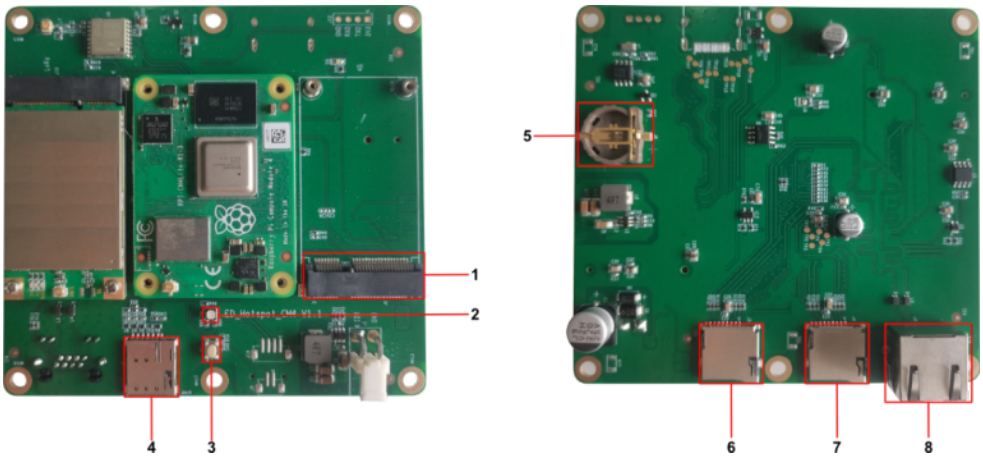
Function	Description
CPU	Broadcom BCM2711, quad core Arm Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memory	Options for 1GB, 2GB, 4GB, 8GB LPDDR4-3200 SDRAM
SD Card	Options for 32GB and 64GB SD card. It supports booting the OS from SD card.
Ethernet	1 x Gigabit Ethernet, RJ45 connector supporting PoE for Ethernet connectivity and device power supply.
WiFi/BT (optiona)	• 2.4GHz and 5GHz dual Wi-Fi, compatible with IEEE 802.11 b/g/n/ac • Bluetooth 5.0, support BLE
4G (optional)	Support various 4G LTE modules
LoRa	Compatible with LoRaWAN protocol, support 3 frequency bands. • 868MHz (EU868) • 915MHz (US915) • 470MHz (CN470)

Function	Description
GNSS	Built-in GNSS functionality, supports multiple satellite systems; compatible with LoRaWAN protocol, supports three frequency bands. • GPS L1 C/A：1575.42±1.023 MHz • BeiDou B1I：1561.098±2.046 MHz • GLONASS L1：1597.78~1605.66 MHz
Internal IO	• 1 x Serial (TTL)，available for the system default console • 1 x User-defined button • 1 x RGB 3-color LED • 1 x RTC battery base, using for installing CR1220 battery • 1 x Nano SIM card slot • 2 x Micro SD card slots
Expansion Functions	• Supports hardware watchdog function • Built-in encryption chip • Built-in RTC
Power Input	PoE power supply, support 802.3af standard
Dimensions	194.2mm (W) x 194.2mm (D) x 65 mm (H)
Case	Cast aluminum waterproof shell, IP65 waterproof grade
Working Temperature	-25°C ~ 60°C

1.3 System Diagram



1.4 Internal IO

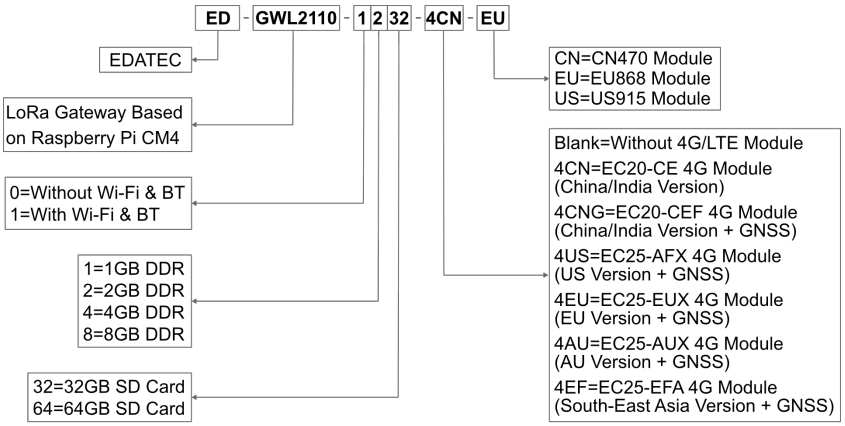


NO.	Function Description
1	4G mini-PCIe Connector
2	RGB LED
3	User Button
4	Nano SIM Card Slot
5	RTC Battery Base
6	Micro SD Card Slot (User Data Storage)
7	Micro SD Card Slot (booting OS)
8	Gigabit Ethernet

1.5 Packing List

- 1 x ED-GWL2110 Unit
- 1 x LoRa Antenna
- [optional Wi-Fi/BT Version] 1x 2.4GHz/5GHz Wi-Fi/BT Antenna
- [optional 4G Version] 1x 4G/LTE Antenna

1.6 Ordering Code



Example

P/N: **ED-GWL2110-1232-4CN-EU**
Configuration: An outdoor LoRa Gateway Based on Raspberry Pi CM4, with Wi-Fi & Bluetooth, 2GB DDR, 32GB SD Card, 4G(EC20-CE Module) and EU868 LoRa Module.

2 Quick Start

This chapter describes the startup and some of the power-up settings for the ED-GWL2110 series.

2.1 Equipment List

- 1 x ED-GWL2110 Unit
- 1 x Wi-Fi/BT External Antenna (optional)
- 1 x LoRa External Antenna
- 1 x 4G External Antenna (optional)
- 1 x Network Cable

2.2 Hardware Connection

1. Install the external antennas for Wi-Fi, LoRa, or 4G separately.
2. Connect the Ethernet cable:
   - One end connects to the device's Ethernet port.
   - The other end connects to a PoE-enabled router or switch with internet access.

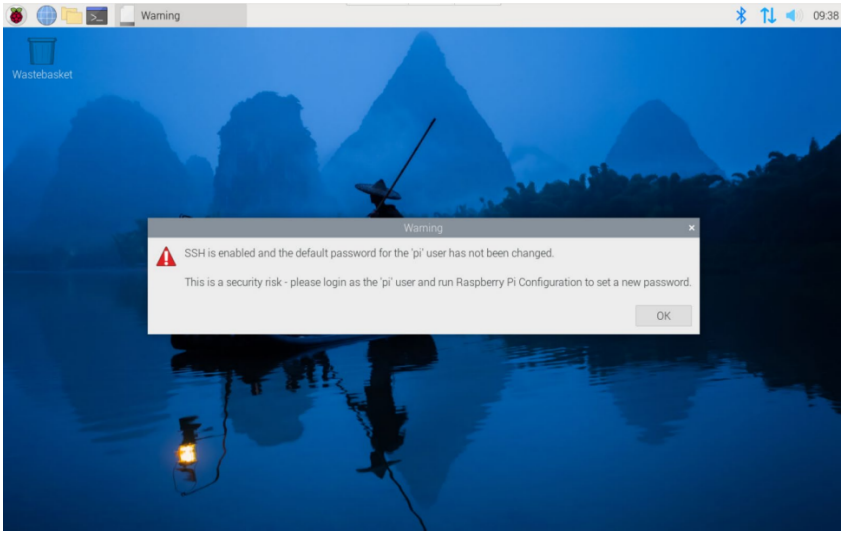
2.3 First Startup

ED-GWL2110 has no power switch. After connecting to PoE power supply, the system will start.

> TIP
>
> - Default username is `pi`, Default password is `raspberry`.
> - The default operating system has SSH enabled.

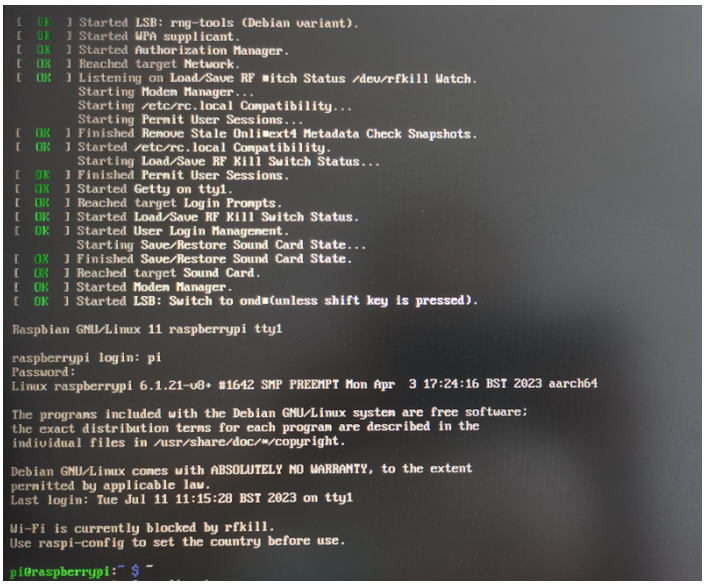
2.3.1 Raspberry Pi OS (Desktop)

If the Desktop version of the system is installed when the product leaves the factory, after the device is started, it will directly enter the desktop, as shown in the following figure.



2.3.2 Raspberry Pi OS (Lite)

If the Lite version of the system is installed at the factory, the default username `pi` will be used to automatically log in after the device is started, and the default password is `raspberry`. The following figure shows that the system has been started normally.



2.4 Finding Device IP

2.4.1 Login Router to Query IP

When the device starts normally but the display is not connected, you can log in to the router to check the current device IP.

Preparation:

- The device has been connected to the network through the router.
- The IP and network password of the router in the network have been obtained, and the IP address is 192.168.X.X.

Steps:

1. Open a browser, Enter the router IP of the network where the device is located in the address bar: 192.168.x.x, and press Enter to enter the router login interface.
2. According to the interface prompts, enter the network password and enter the router management interface.
3. Find the IP address of device in the terminal device of the management interface.

2.4.2 Scan For Using NMAP Tool

When the device starts normally but the display is not connected, you can use nmap tool to scan the IP under the current network to obtain the IP information of the device. Nmap supports Linux, macOS, Windows and other platforms.

Preparation:

- The device has been connected to the network through the router.
- The IP segment and mask of the router in the network have been obtained, for example, 192.168.X.X/24, where 24 is the subnet mask.

Steps:

For example, using nmap to scan the network segments from 192.168.3.0 to 255, you can use the following steps:

1. Open the nmap tool and scan the hosts in the 192.168.X.X/24 network segment.

> WARNING
>
> The nmap tool operates differently in different operating systems, so please follow the actual interface or command prompts.

2. According to the scanned results, get the device IP.

2.5 Connect To The Device Via SSH

After the device starts normally, you can choose to connect to the device remotely through SSH to configure or debug it. The tools for remote login are selected by users themselves, and the following is an example of logging in through MobaXterm.

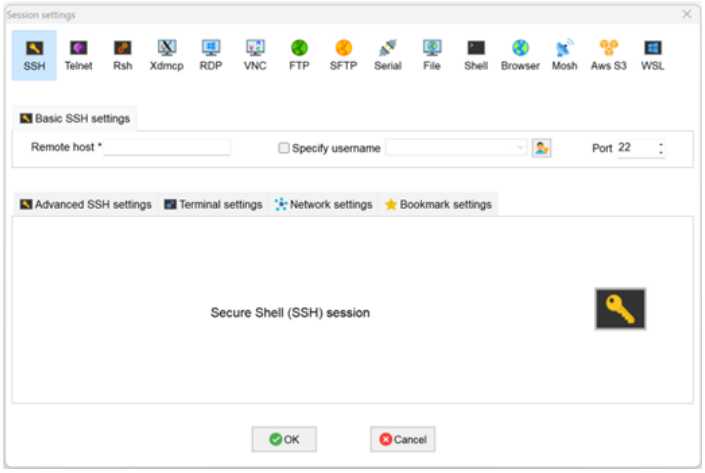
Preparation:

- The MobaXterm tool has been installed on the PC.
- The device has been connected to the network through the router.
- IP address of the device has been get.

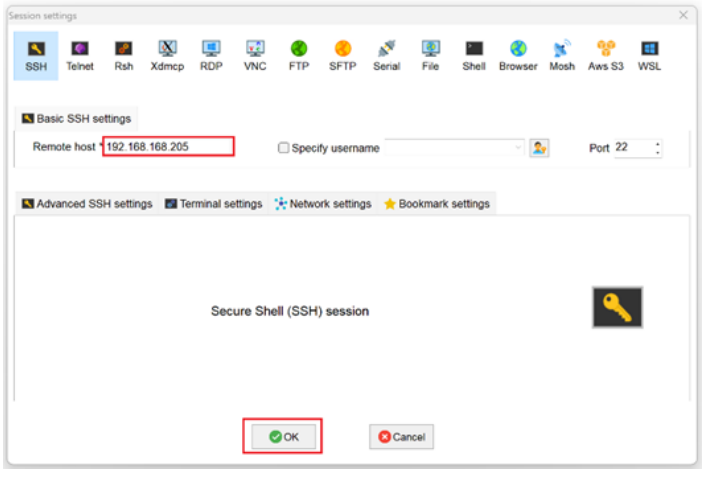
Steps:

1. Open MobaXterm, click Session, and open the window for creating connection, as shown in the figure below.

2. Click SSH in the upper left corner to open the SSH connection interface.



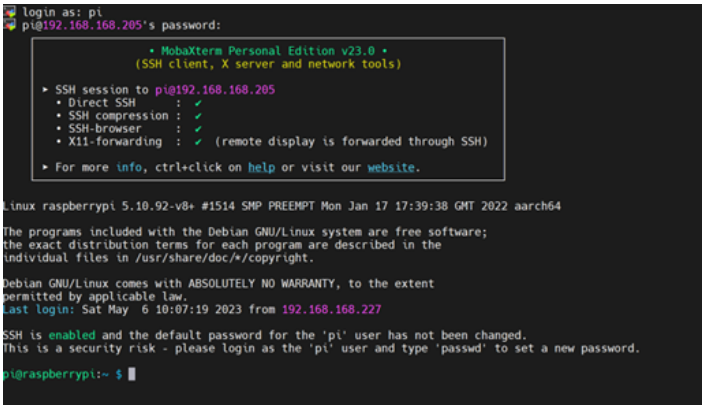
3. After entering the IP address of the obtained device, click "OK".



4. Click "Accept" in the pop-up prompt box to enter the system login interface.

5. Enter the username and password according to the prompt, and enter the system after logging in.

> TIP
>
> Default username is pi, Default password is raspberry.



3 Software Operation Guide

3.1 Button

The D-GWL2110 includes a user-defined button located internally, connected to the CPU's GPIO23 pin. By default, the pin is in a high-level state, which switches to a low-level state when the button is pressed.

You can use the `raspi-gpio` command to check the state of GPIO.

- Query the GPIO23 pin when the button is not pressed.

```sh
raspi-gpio get 23
GPIO 23: level=1 fsel=0 func=INPUT
```

Level of 1 indicates that the GPIO23 pin is high.

- Query the GPIO23 pin When the button is pressed.

```sh
raspi-gpio get 23
GPIO 23: level=0 fsel=0 func=INPUT
```

Level of 0 indicates that the GPIO23 pin is low.

3.2 LED Indicator

ED-GWL2110 includes an RGB 3-color LED indicator, and the corresponding GPIO pins are as follows:

RGB LED PIN	GPIO
Blue	GPIO16 11
Green	GPIO20 12
Red	GPIO21 13

When the GPIO output is low, the corresponding LED is active.

You can use the `raspi-gpio` command to check the state of GPIO.

- Configuration parameter "op": Sets as output;
- "dl": Sets the device pin to low level;
- "dh": Sets the pin to high level.

The LED is displayed in blue.

```sh
sudo raspi-gpio set 11 op dl
sudo raspi-gpio set 12 op dh
sudo raspi-gpio set 13 op dh
```

The LED is displayed in green.

```sh
sudo raspi-gpio set 11 op dh
sudo raspi-gpio set 12 op dl
sudo raspi-gpio set 13 op dh
```

The LED is displayed in red.

```sh
sudo raspi-gpio set 11 op dh
sudo raspi-gpio set 12 op dh
sudo raspi-gpio set 13 op dl
```

The LED is displayed in yellow.

```sh
sudo raspi-gpio set 11 op dh
sudo raspi-gpio set 12 op dl
sudo raspi-gpio set 13 op dl
```

3.3 Configuring Ethernet IP

The IP address is automatically obtained by default. If you need to reconfigure the IP, you can configure it through NetworkManager.

> TIP
>
> The factory-default Lite operating system has NetworkManager enabled, allowing direct configuration via NetworkManager.

3.3.1 Raspberry Pi OS (Desktop)

In the Desktop version of the operating system, it is recommended to use the graphical NetworkManager tool to configure IP.

> WARNING

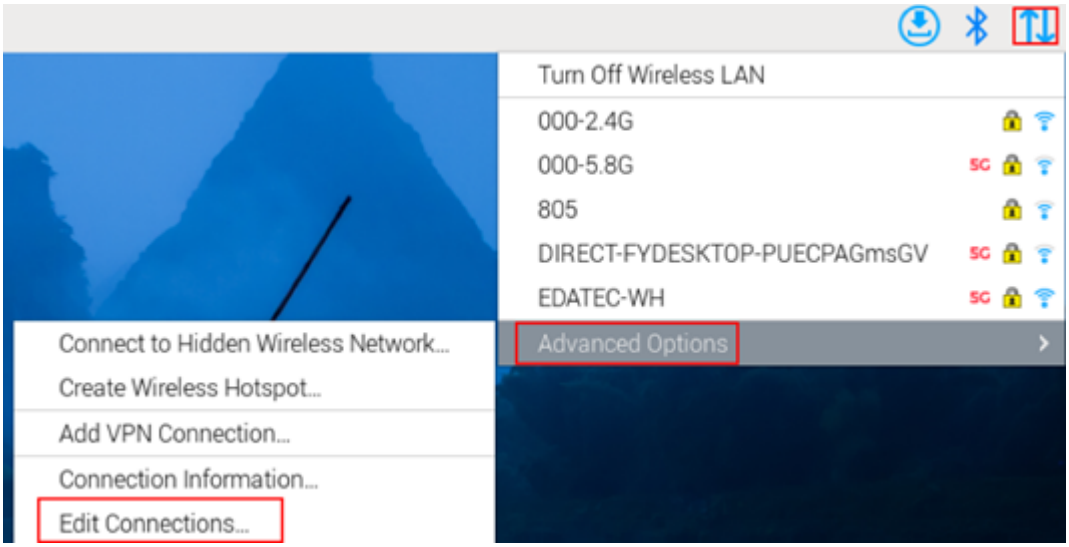
> The Desktop version of the operating system has the NetworkManager graphical tool installed by default.

Preparation:

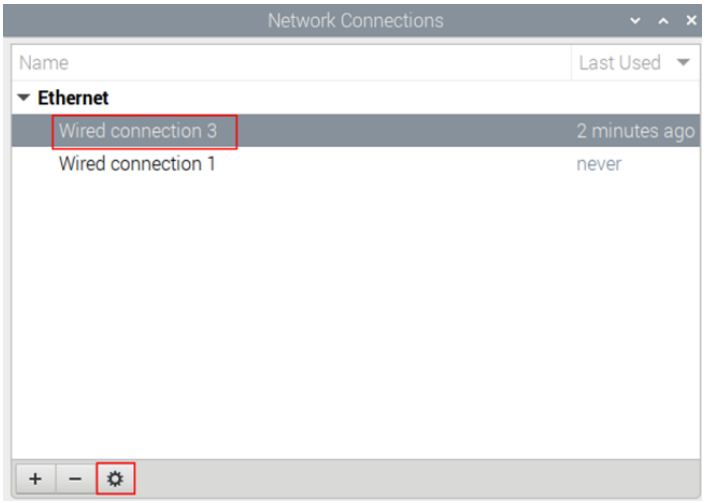
Wi-Fi is enabled.

Steps:

1. Left-click the icon in the upper right corner of the desktop and select “Advanced Options→Edit Connections” in the pop-up menu.

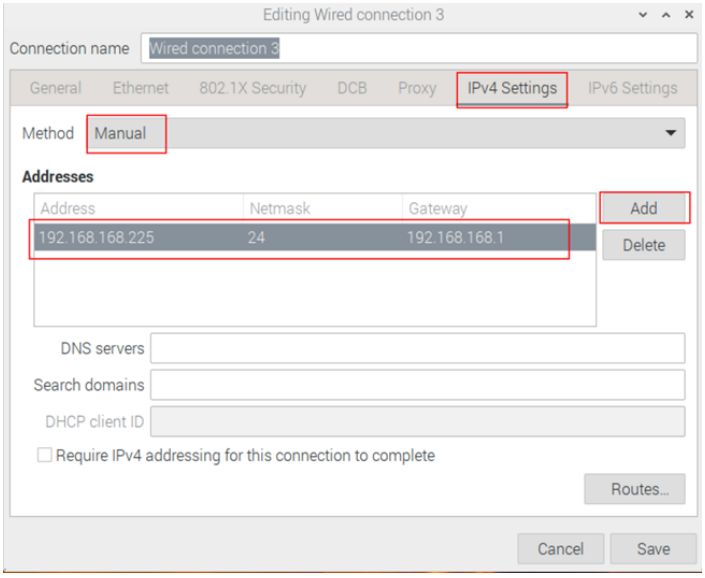


2. In the pop-up "Network Connections" pane, select the connection name to be modified, and then click the Settings button below.

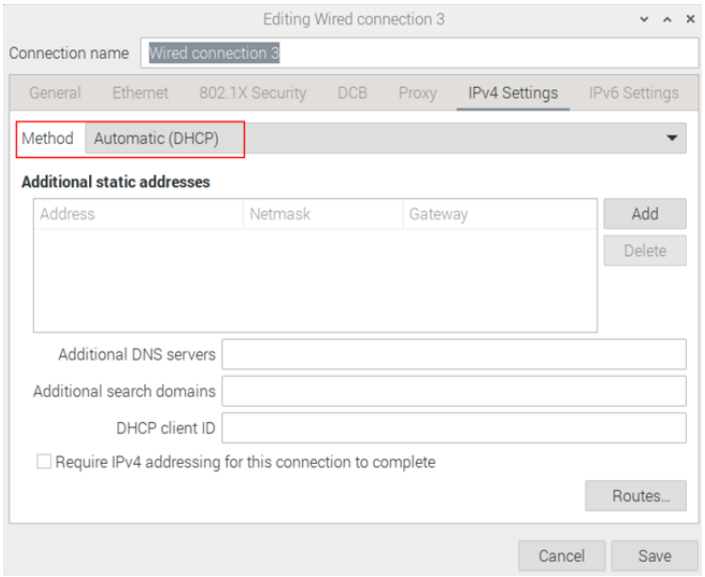


3. In the pop-up "Editing Wired connection" pane, select the "IPv4 Settings" page, and then set the IP address as required.

- If you want to set the IP as a static IP, set the "Method" as "Manual", add an entry in Addresses and enter the corresponding IP address information.



- If you want to set the IP to automatic mode, you only need to set the "Method" as "Automatic(DHCP) ".



4. Click "save" to return to "Network Connections" pane and close the page.
5. Execute the sudo reboot command to restart the device.

3.3.2 Raspberry Pi OS (Lite)

In the Lite version of operating system, it is recommended to use the command to configure IP.

Preparation:

NetworkManager is enabled.

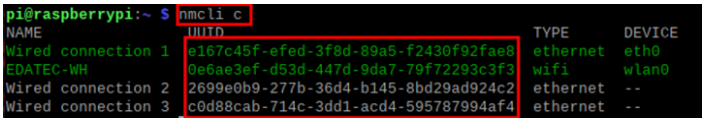
Steps:

Set a static IP address

1. Get the assigned IP address, subnet mask and gateway address, for example, the IP address is 192.168.1.101/24 and the gateway IP is 192.168.1.1.

2. Obtain the connection name to be modified, for example `e167c45f-efed-3f8d-89a5-f2430f92fae8`. In the command pane, run the following command to query the connection name.

```sh
nmcli c
```



3. Execute the following command to set the IP address to the obtained IP address.

```sh
sudo nmcli connection modify e167c45f-efed-3f8d-89a5-f2430f92fae8 ipv4.addresses 192.168.1.101
```

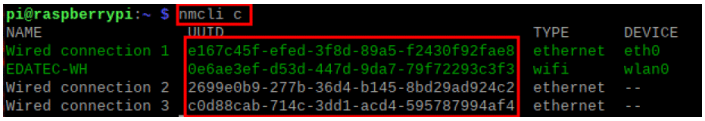
4. Execute the following command to set the gateway IP to the obtained gateway IP.

```sh
sudo nmcli connection modify e167c45f-efed-3f8d-89a5-f2430f92fae8 ipv4.gateway 192.168.1.1
```

Set the IP to automatic mode

1. Obtain the connection name to be modified, for example e167c45f-efed-3f8d-89a5-f2430f92fae8. In the command pane, run the following command to query the connection name.

```sh
nmcli c
```



2. Execute the following command to set the way of obtaining IP address to automatic mode.

```sh
sudo nmcli connection modify e167c45f-efed-3f8d-89a5-f2430f92fae8 ipv4.method auto
```

3.4 Configuring Wi-Fi (Optional)

Configuring Wi-Fi

3.5 Configuring Bluetooth (Optional)

Configuring Bluetooth

3.6 Configuring 4G (Optional)

Configuring 4G

3.7 Configuring Storage Devices (SD Card)

Configuring Storage Devices

3.8 Configuring RTC

Configuring RTC

3.9 GNSS

ED-GWL2110 gateway integrates L76K GPS module, which is connected with UART0 serial port of CPU. The module reports GNSS information through NMEA 0183 general protocol output statement.

3.9.1 Pin Configuration

The WakeUp signal of L76K GPS module is connected to GPIO4. If the pin module is pulled down, it will enter standby mode, and if it is pulled up or suspended, it will return to continuous mode. The Reset signal is connected to GPIO5. Pulling this pin low for 100ms will reset the module. SET signal is connected with GPIO6, which is used to configure the satellite combination. When the pin is suspended or high level, the satellite combination is GPS and Beidou, and when the pin is low level, the satellite combination is GPS and GLONASS.

- The L76K GPS module's WakeUp signal is connected to GPIO4. Pulling this pin low will put the module into standby mode, while pulling it high or leaving it floating will return it to continuous mode.
- The Reset signal is connected to GPIO5. Pulling this pin low and holding it for at least 100ms will reset the module.
- The SET signal is connected to GPIO6 and is used to configure the satellite combination:
  - When the pin is floating or pulled high, the satellite combination is GPS + Beidou;
  - When the pin is pulled low, the satellite combination is GPS + GLONASS.

Pin	Signal	CM4 Pinout
1	GPS_WakeUp	GPIO4
2	GPS_Reset	GPIO5
3	GPS_Set	GPIO6

3.9.2 Modify `config.txt` to Enable Serial Port

1. Execute the following command to open the `config.txt` file.

```sh
sudo nano /boot/config.txt
```

2. Add "enable_uart=1" at the end of the file.
3. Press `Ctrl+O` to save the configuration file, then press `Enter`, and finally press `Ctrl+X to` exit the editor.

3.9.3 Check GNSS information

```sh
sudo cat /dev/ttyS0
```

Display GPS data as follows:

```text
$BDGSV,3,1,11,04,29,117,20,10,,,19,16,75,160,,24,51,328,,0*4C
$BDGSV,3,2,11,25,,,27,26,,,21,34,12,198,,35,45,063,,0*76
$BDGSV,3,3,11,39,62,159,17,41,,,25,59,44,137,,0*7A
$GNRMC,053557.000,A,3027.47401,N,11424.34027,E,1.17,186.64,070223,,,A,V*05
$GNVTG,186.64,T,,M,1.17,N,2.17,K,A*2D
$GNZDA,053557.000,07,02,2023,00,00*4F
$GPTXT,01,01,01,ANTENNA OPEN*25
$GNGGA,053558.000,3027.47438,N,11424.34119,E,1,07,1.5,75.0,M,-14.1,M,,*52
$GNGLL,3027.47438,N,11424.34119,E,053558.000,A,A*4F
$GNGSA,A,3,07,08,16,31,195,,,,,,,2.1,1.5,1.5,1*05
$GNGSA,A,3,04,39,,,,,,,,,,,2.1,1.5,1.5,4*39
$GPGSV,3,1,12,04,54,241,16,07,19,314,15,08,63,208,15,09,38,291,,0*67
$GPGSV,3,2,12,16,51,029,17,18,07,046,,21,08,175,,26,24,063,,0*6A
$GPGSV,3,3,12,27,77,065,,31,09,122,22,194,61,058,,195,46,125,21,0*66
```

NMEA 0183 general statement is described as follows:

- BDGSV: Visible Beidou satellite information
- GNRMC: Recommended GNSS data
- GNVTG: Relative ground heading and speed information
- GNZDA: Time and date, UTC format
- GPTXT: Text transmission
- GNGGA: Multi-satellite joint positioning data
- GNGLL: Geographical location, latitude and longitude
- GNGSA: Represents GNSS accuracy factor and effective satellite.
- GPGSV: Visible GNSS satellite

3.9.4 Use the u-center tool to view positioning information

3.9.4.1 Install serial port to network tool ser2net

```sh
sudo apt-get update
sudo apt-get install ser2net
```

Enable ser2net service

Ser2net configuration file is `/etc/ser2net.yaml`. By default, `/dev/ttyS0` is configured, baud rate is 9600, and there is no check, and the corresponding TCP port is 2000.

```sh
connection: &con0096
    accepter: tcp,2000
    enable: on
    options:
      banner: *banner
      kickolduser: true
      telnet-brk-on-sync: true
    connector: serialdev,
              /dev/ttyS0,
              9600n81,local
```

3.9.4.2 Check ser2net Port Forwarding Service

Execute the following command to query whether ser2net has started 2000 port forwarding.

```sh
sudo netstat -ltnp | grep 2000
```

- If the port forwarding has been started, the following message will be displayed.

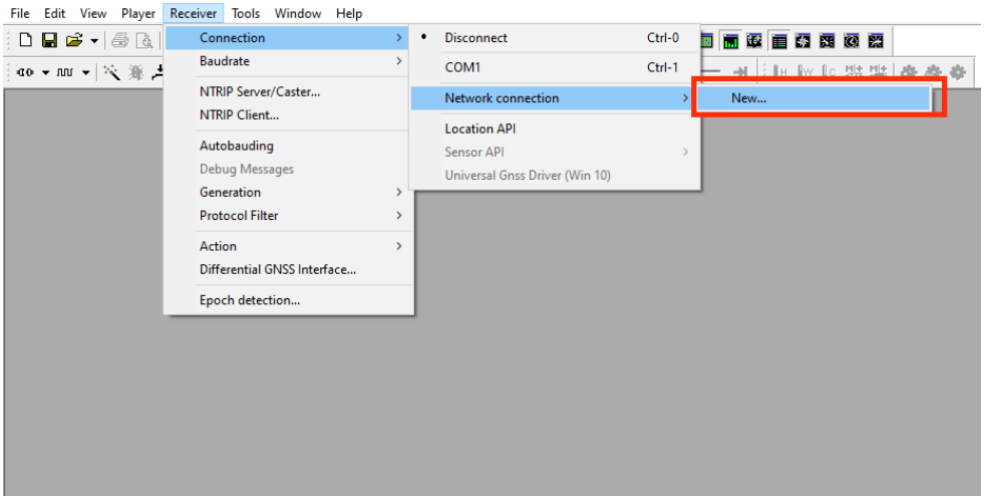
```text
tcp6    0    0 :::2000         :::*          LISTEN    720/ser2net
```

- If not, restart the ser2net service.

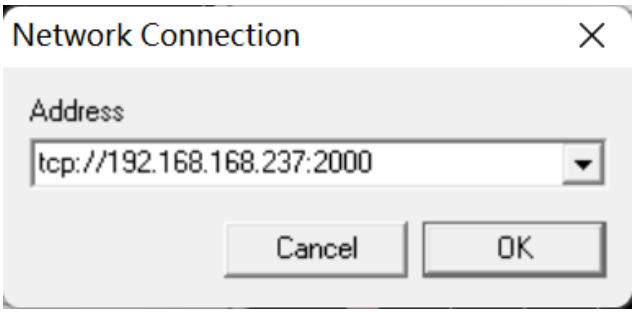
```sh
sudo systemctl restart ser2net
```

Positioning Configuration

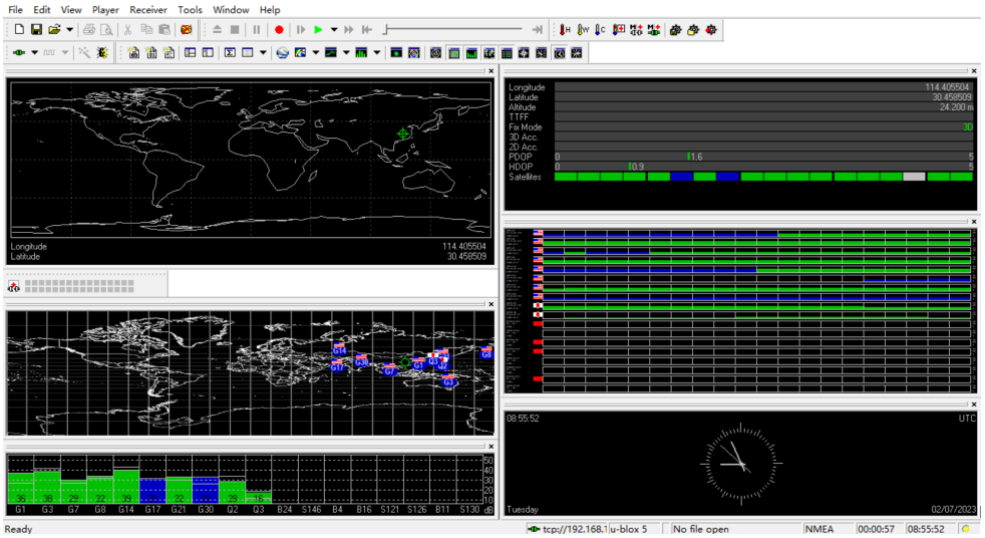
1. Download and install the u-center (https://www.u-blox.com/en/product/u-center) tool. If you are prompted that the `MSVCR120.dll` file is missing, please install vcredist_x86.exe (https://www.microsoft.com/en-us/download/details.aspx?id=40784) .
2. Open `u-center` , Choose "Receiver"→"Port"→"Network connection"→"New…".



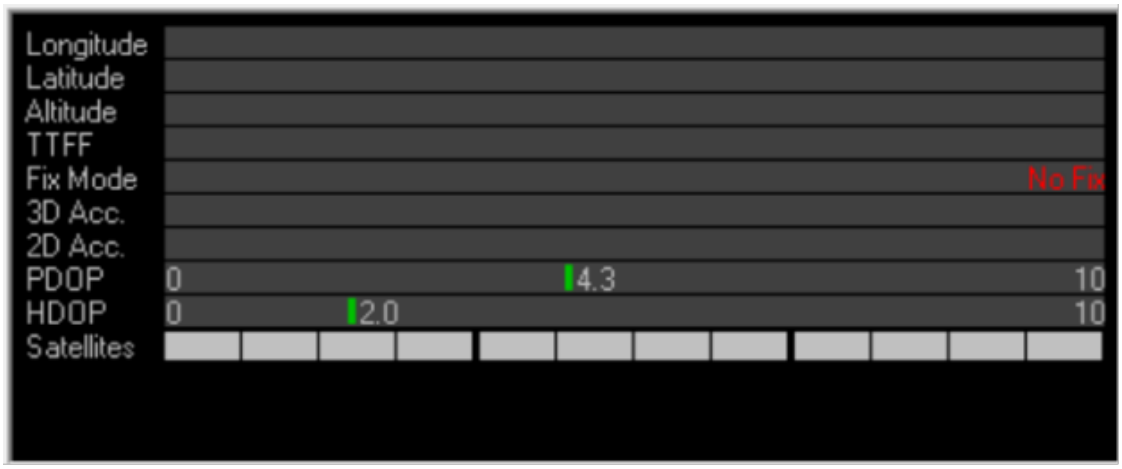
3. Enter your device IP and port number 2000.



4. You will see the GPS positioning information immediately after the configuration is completed.



- If the Fix Mode is displayed as No Fix, it means that the positioning failed, which is usually caused by the antenna being indoors. Please put the module or antenna outdoors for testing.



> NOTE
>
> For the first time, it takes about 30 seconds to locate the module successfully when there is no large building outside. If the weather conditions are bad, it may take longer or it may not be possible to locate it.

3.10 LoRaWAN

The ED-GWL2110 supports LoRaWAN and the open-source service platform ChirpStack. The following section details the specific installation and configuration procedures.

> TIP
>
> Only the US915 LoRa gateway module is used as an example for this introduction.

3.10.1 Chirpstack Client

- The device comes pre-installed with firmware packages, LoRa module packages, and ChirpStack software packages. Verify using:

```sh
dpkg -l | grep ed-
```

- For OS reinstallation, refer to the OS Installation section.

3.10.2 Configuration File Modifications

3.10.2.1 Modify JSON Configuration File

1. Start `ed-lora.service`:

```sh
sudo systemctl enable --now ed-lora.service
```

2. Get the configuration file path:

```sh
systemctl status ed-lora.service
```

```
pi@GWL[redacted]:~ $ systemctl status ed-lora.service
● ed-lora.service - EDATec LoRa Packet Forwarder Service
     Loaded: loaded (/lib/systemd/system/ed-lora.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-03-04 11:12:18 CST; 5s ago
   Main PID: 2447 (start.sh)
      Tasks: 5 (limit: 8731)
        CPU: 174ms
     CGroup: /system.slice/ed-lora.service
             ├─2447 /bin/bash /opt/ed-lora/start.sh
             └─2452 /opt/ed-lora/lora_pkt_fwd -c /opt/ed-lora/conf/global_conf.json.US915

Mar 04 11:12:18 GWL2010 systemd[1]: Started ed-lora.service - EDATec LoRa Packet Forwarder Service.
Mar 04 11:12:18 GWL2010 start.sh[2447]: 2025-03-04 11:12:18 [info] pktfwd: started with conf file: /opt/ed-lora/conf/global_conf.json.US915
Mar 04 11:12:18 GWL2010 start.sh[2452]: ERROR: failed to configure uart port
Mar 04 11:12:19 GWL2010 start.sh[2454]: CoreCell reset through GPIO18...
```

3. Open the configuration file:

```sh
sudo nano /opt/ed-lora/conf/global_conf.json.US915
```

4. Delete `"gps_i2c_path": "/dev/i2c-1"` at the bottom.

```
            "bandwidth": 250000,
            "datarate": 100000
        }
    },
    "gateway_conf": {
        "gps_i2c_path": "/dev/i2c-1",
        "gateway_ID": "AA555A0000000000",
        /* change with default server address/ports */
        "server_address": "localhost",
        "serv_port_up": 1680,
        "serv_port_down": 1680,
        /* adjust the following parameters for your network */
        "keepalive_interval": 10,
        "stat_interval": 30,
        "push_timeout_ms": 100,
        /* forward only valid packets */
        "forward_crc_valid": true,
        "forward_crc_error": false,
        "forward_crc_disabled": false
    }
}
```

3.10.2.2 Modify ChirpStack Configuration

1. Open the configuration file.

```sh
sudo nano /etc/chirpstack-gateway-bridge/chirpstack-gateway-bridge.toml
```

2. Make the following changes:
   - `udp_bind = "0.0.0.0:1700"` → `udp_bind = "0.0.0.0:1680"`
   - Add gateway model prefix:
     - `event_topic_template="gateway/{{ .GatewayID }}/event/{{ .EventType }}"` → `event_topic_template="us915_0/gateway/{{ .GatewayID }}/event/{{ .EventType }}"`.
     - `command_topic_template="gateway/{{ .GatewayID }}/command/#"` → `command_topic_template="us915_0/gateway/{{ .GatewayID }}/command/#"`.

> TIP
>
> For EU868/CN470 modules, replace `us915_0` with `eu868_0` or `cn470_10`.

3. Restart services:

```sh
sudo systemctl daemon-reload
sudo systemctl restart chirpstack-gateway-bridge.service ed-lora.service
```

3.10.3 ChirpStack Server Deployment

The ChirpStack server can be deployed either on the current device or on another device within the same LAN. This documentation uses deployment on the current device as an example. (If deploying on another LAN device, modify the `udp_bind` value from `0.0.0.0` to that device's IP address in Section 3.10.2.2 Modify ChirpStack Configuration).

1. Install dependencies:

```sh
sudo apt install docker-compose -y
```

2. Clone the repository:

```sh
git clone https://github.com/chirpstack/chirpstack-docker.git
```

```
pi@GWL      :~ $ git clone https://github.com/chirpstack/chirpstack-docker.git
Cloning into 'chirpstack-docker'...
remote: Enumerating objects: 572, done.
remote: Counting objects: 100% (413/413), done.
remote: Compressing objects: 100% (132/132), done.
remote: Total 572 (delta 357), reused 285 (delta 281), pack-reused 159 (from 4)
Receiving objects: 100% (572/572), 101.75 KiB | 166.00 KiB/s, done.
Resolving deltas: 100% (389/389), done.
```

3. Modify configuration:

```sh
cd chirpstack-docker
nano docker-compose.yml
```

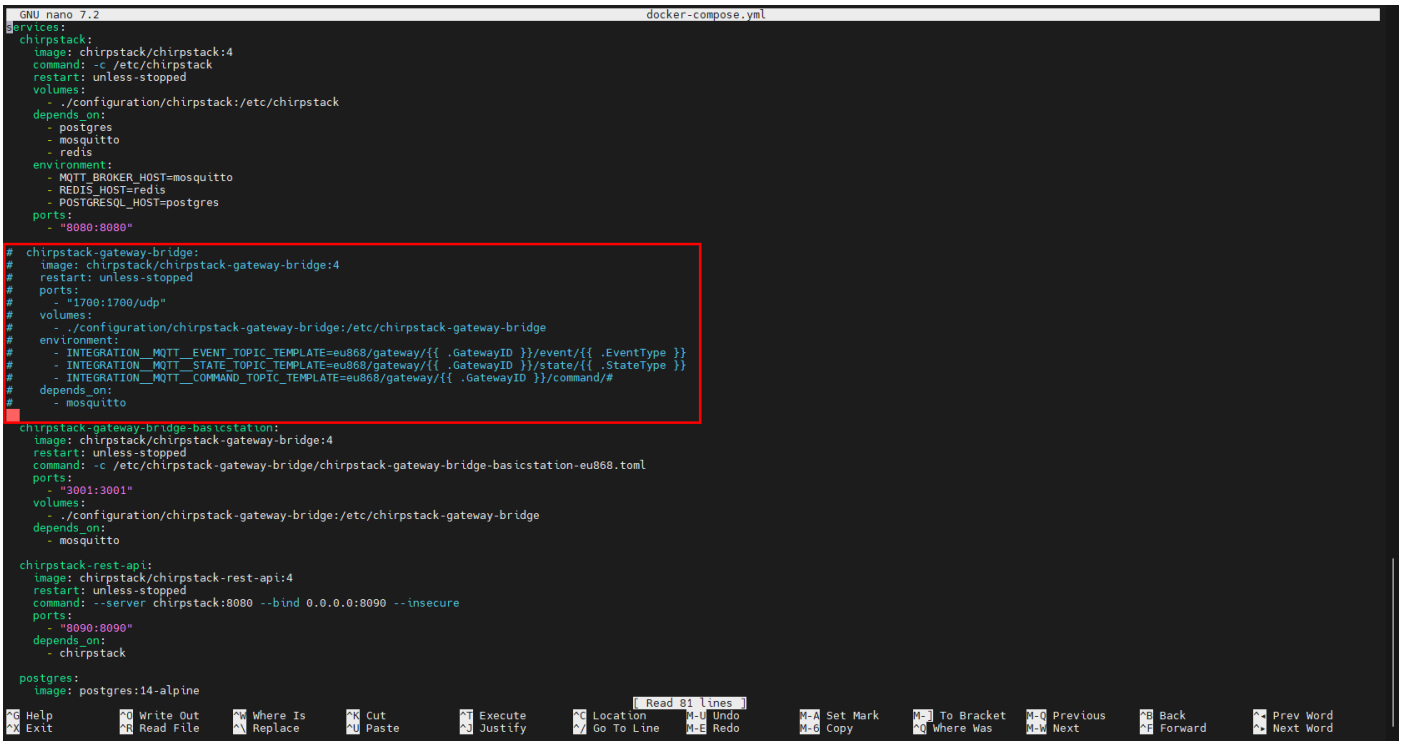
4. Comment out the `chirpstack-gateway-bridge` section.

```sh
services:
  chirpstack:
    image: chirpstack/chirpstack:4
    command: -c /etc/chirpstack
    restart: unless-stopped
    volumes:
      - ./configuration/chirpstack:/etc/chirpstack
    depends_on:
      - postgres
      - mosquitto
      - redis
    environment:
      - MQTT_BROKER_HOST=mosquitto
      - REDIS_HOST=redis
      - POSTGRESQL_HOST=postgres
    ports:
      - "8080:8080"

#  chirpstack-gateway-bridge:
#    image: chirpstack/chirpstack-gateway-bridge:4
#    restart: unless-stopped
#    ports:
#      - "1700:1700/udp"
#    volumes:
#      - ./configuration/chirpstack-gateway-bridge:/etc/chirpstack-gateway-bridge
#    environment:
#      - INTEGRATION__MQTT__EVENT_TOPIC_TEMPLATE=eu868/gateway/{{ .GatewayID }}/event/{{ .Even
#      - INTEGRATION__MQTT__STATE_TOPIC_TEMPLATE=eu868/gateway/{{ .GatewayID }}/state/{{ .Stat
#      - INTEGRATION__MQTT__COMMAND_TOPIC_TEMPLATE=eu868/gateway/{{ .GatewayID }}/command/#
#    depends_on:
#      - mosquitto

  chirpstack-gateway-bridge-basicstation:
    image: chirpstack/chirpstack-gateway-bridge:4
    restart: unless-stopped
```

```
    command: -c /etc/chirpstack-gateway-bridge/chirpstack-gateway-bridge-basicstation-eu868.to
    ports:
      - "3001:3001"
    volumes:
      - ./configuration/chirpstack-gateway-bridge:/etc/chirpstack-gateway-bridge
    depends_on:
      - mosquitto

  chirpstack-rest-api:
    image: chirpstack/chirpstack-rest-api:4
    restart: unless-stopped
    command: --server chirpstack:8080 --bind 0.0.0.0:8090 --insecure
    ports:
      - "8090:8090"
    depends_on:
      - chirpstack

  postgres:
    image: postgres:14-alpine
    restart: unless-stopped
    volumes:
      - ./configuration/postgresql/initdb:/docker-entrypoint-initdb.d
      - postgresqldata:/var/lib/postgresql/data
    environment:
      - POSTGRES_USER=chirpstack
      - POSTGRES_PASSWORD=chirpstack
      - POSTGRES_DB=chirpstack

  redis:
    image: redis:7-alpine
    restart: unless-stopped
    command: redis-server --save 300 1 --save 60 100 --appendonly no
    volumes:
      - redisdata:/data

  mosquitto:
    image: eclipse-mosquitto:2
    restart: unless-stopped
    ports:
      - "1883:1883"
    volumes:
      - ./configuration/mosquitto/config/:/mosquitto/config/

volumes:
  postgresqldata:
  redisdata:
```

TIP

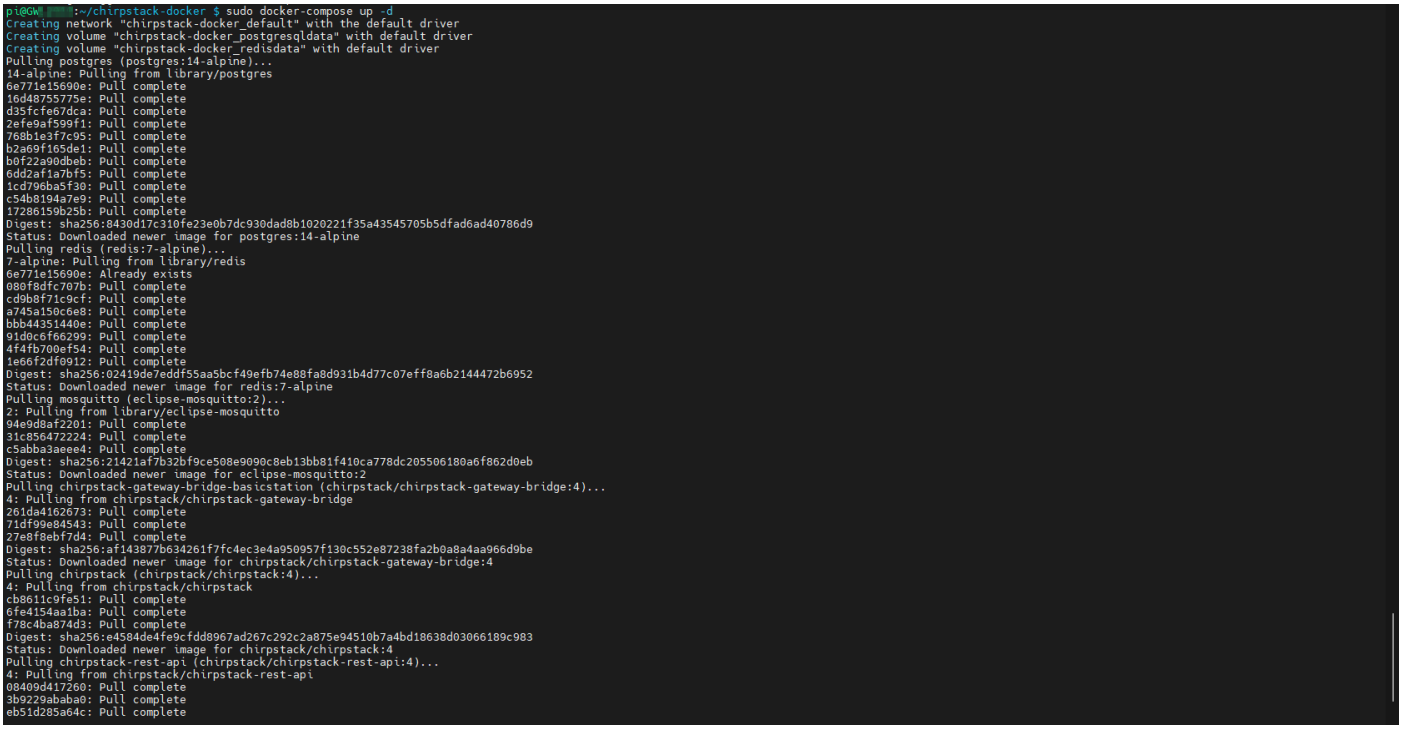
> Customize MQTT service settings in the `mosquitto` section.



3.10.4 Start ChirpStack Server

Execute the following command on the device where the ChirpStack server is deployed to start the ChirpStack server.

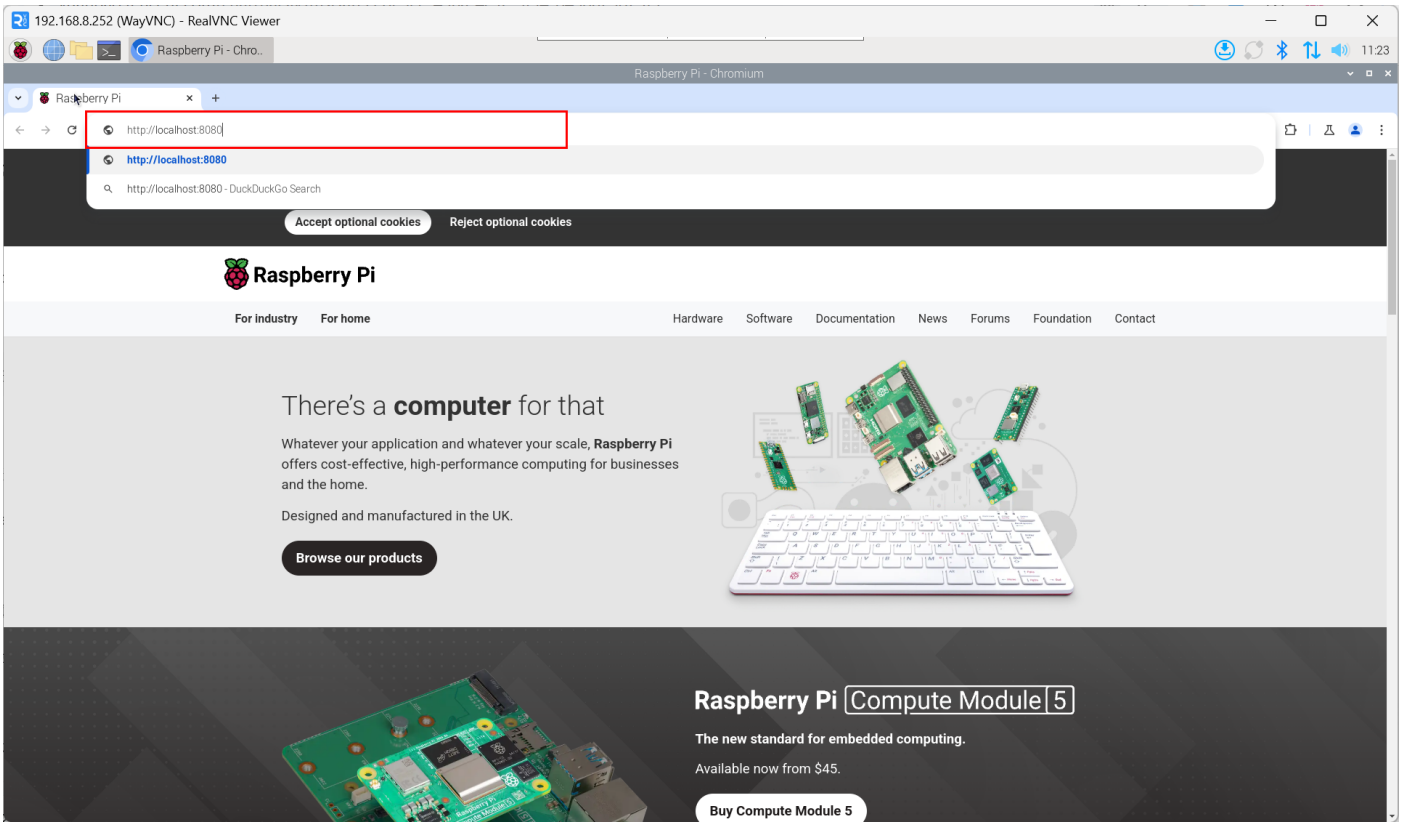
```sh
sudo docker-compose up -d
```



3.10.5 ChirpStack Web Interface Operations

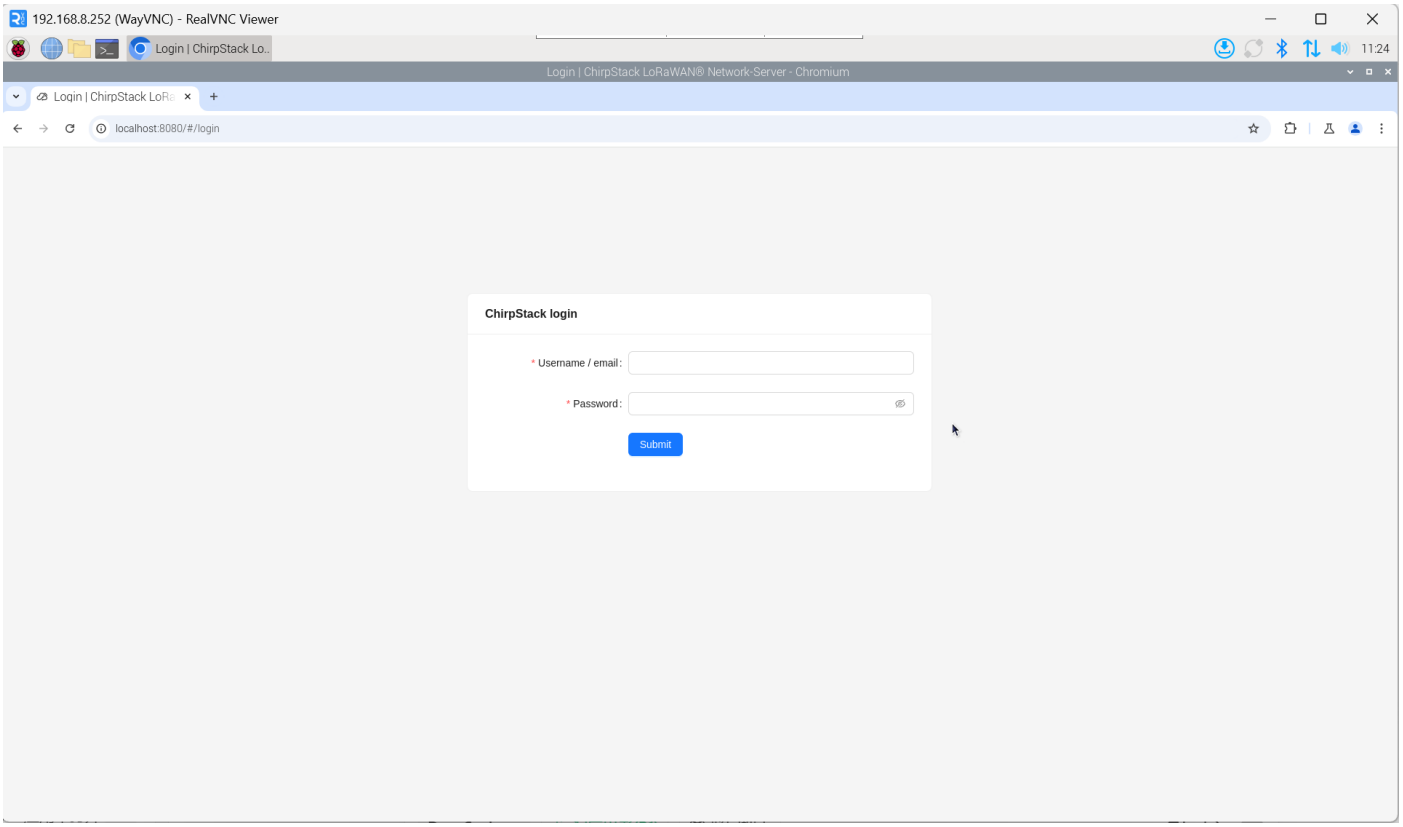
3.10.5.1 Access Web Interface

1. Access via browser: `http://<server_ip>:8080`.



2. Enter the username and password on the login interface.

> TIP
>
> The default username is `admin`, and the default password is `admin`.



3.10.5.2 Obtain Gateway ID

1. Check service status:

```sh
systemctl status ed-lora.service
```

```
pi@GWL:~ $ systemctl status ed-lora.service
● ed-lora.service - EDATec LoRa Packet Forwarder Service
     Loaded: loaded (/lib/systemd/system/ed-lora.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-03-04 11:12:18 CST; 13min ago
   Main PID: 2447 (start.sh)
      Tasks: 5 (limit: 8731)
        CPU: 35.715s
     CGroup: /system.slice/ed-lora.service
             ├─2447 /bin/bash /opt/ed-lora/start.sh
             └─2452 /opt/ed-lora/lora_pkt_fwd -c /opt/ed-lora/conf/global_conf.json.US915

Mar 04 11:24:21 GWL2010 start.sh[2452]: ### Concentrator temperature: 30 C ###
Mar 04 11:24:21 GWL2010 start.sh[2452]: ##### END #####
Mar 04 11:24:21 GWL2010 start.sh[2452]: JSON up: {"stat":{"time":"2025-03-04 03:23:51 GMT","rxnb":0,"rxok":0,"rxfw":0,"ackr":0.0,"dwnb":0,"txnb":0,"temp":30.0}}
Mar 04 11:24:21 GWL2010 start.sh[2452]: ##### 2025-03-04 03:24:21 GMT #####
Mar 04 11:24:21 GWL2010 start.sh[2452]: ### [UPSTREAM] ###
Mar 04 11:24:21 GWL2010 start.sh[2452]: # RF packets received by concentrator: 0
Mar 04 11:24:21 GWL2010 start.sh[2452]: # CRC_OK: 0.00%, CRC_FAIL: 0.00%, NO_CRC: 0.00%
Mar 04 11:24:21 GWL2010 start.sh[2452]: # RF packets forwarded: 0 (0 bytes)
Mar 04 11:24:21 GWL2010 start.sh[2452]: # PUSH_DATA datagrams sent: 1 (123 bytes)
Mar 04 11:24:21 GWL2010 start.sh[2452]: # PUSH_DATA acknowledged: 0.00%
pi@GWL:~ $
```

2. Extract Gateway ID:

```sh
cat /opt/ed-lora/conf/global_conf.json.US915 | grep gateway
```

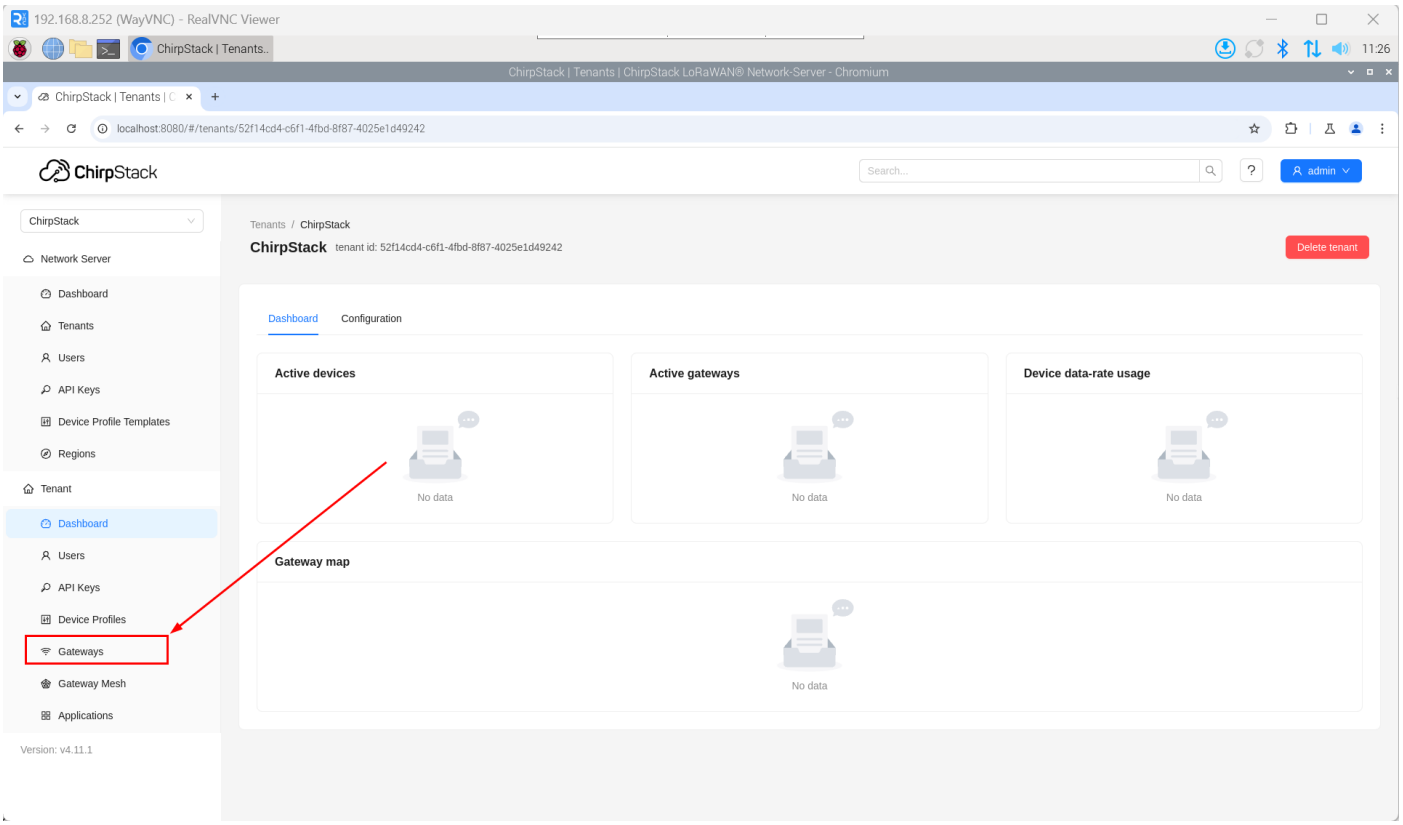
```
pi@GWL:~ $ cat /opt/ed-lora/conf/global_conf.json.US915 | grep gateway
    "gateway_conf": {
        "gateway_ID": "AA555A0000000000",
```

> TIP

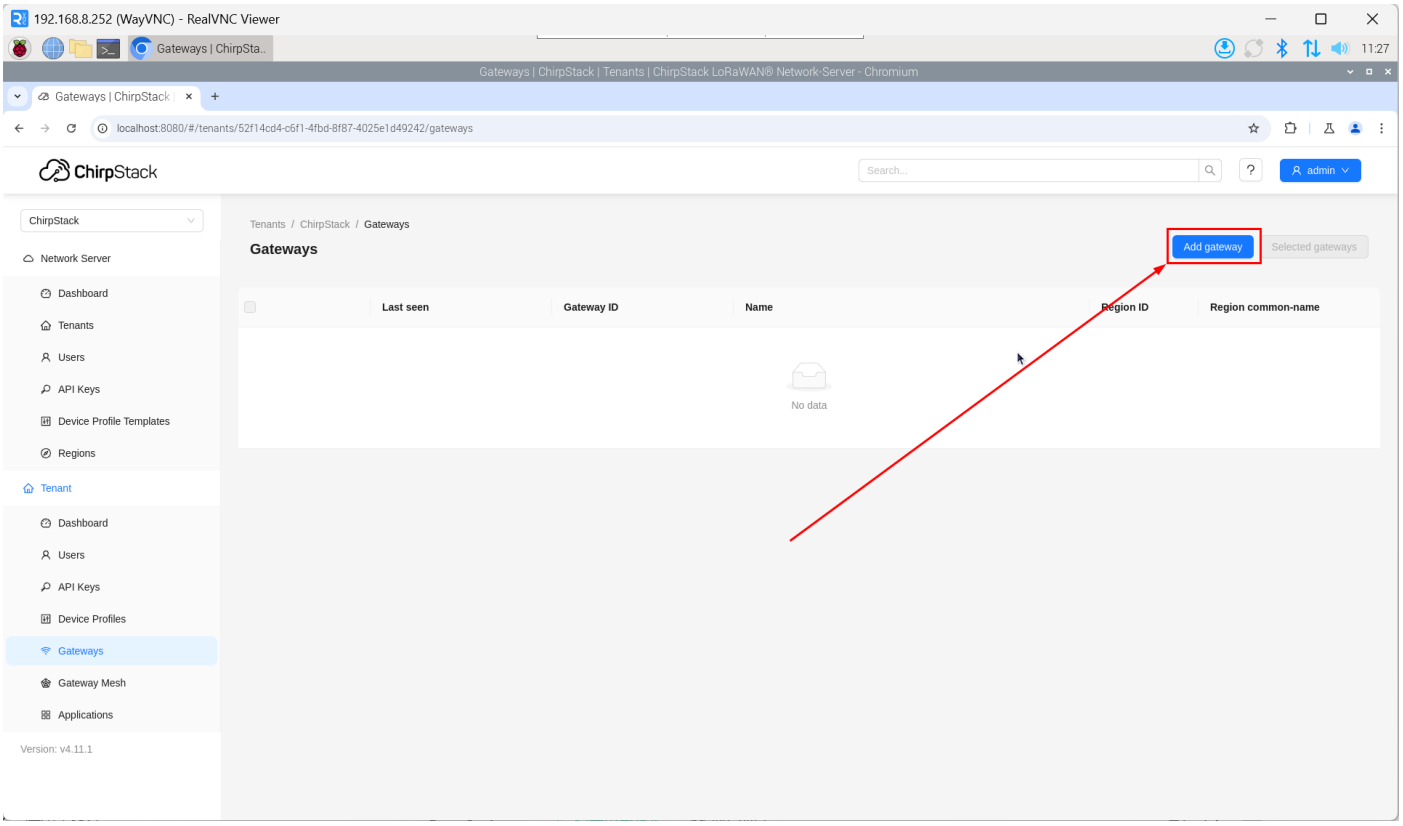
> When adding a LoRa gateway on the ChirpStack server side, you need to provide the corresponding Gateway ID.

3.10.5.3 Add LoRa Gateway

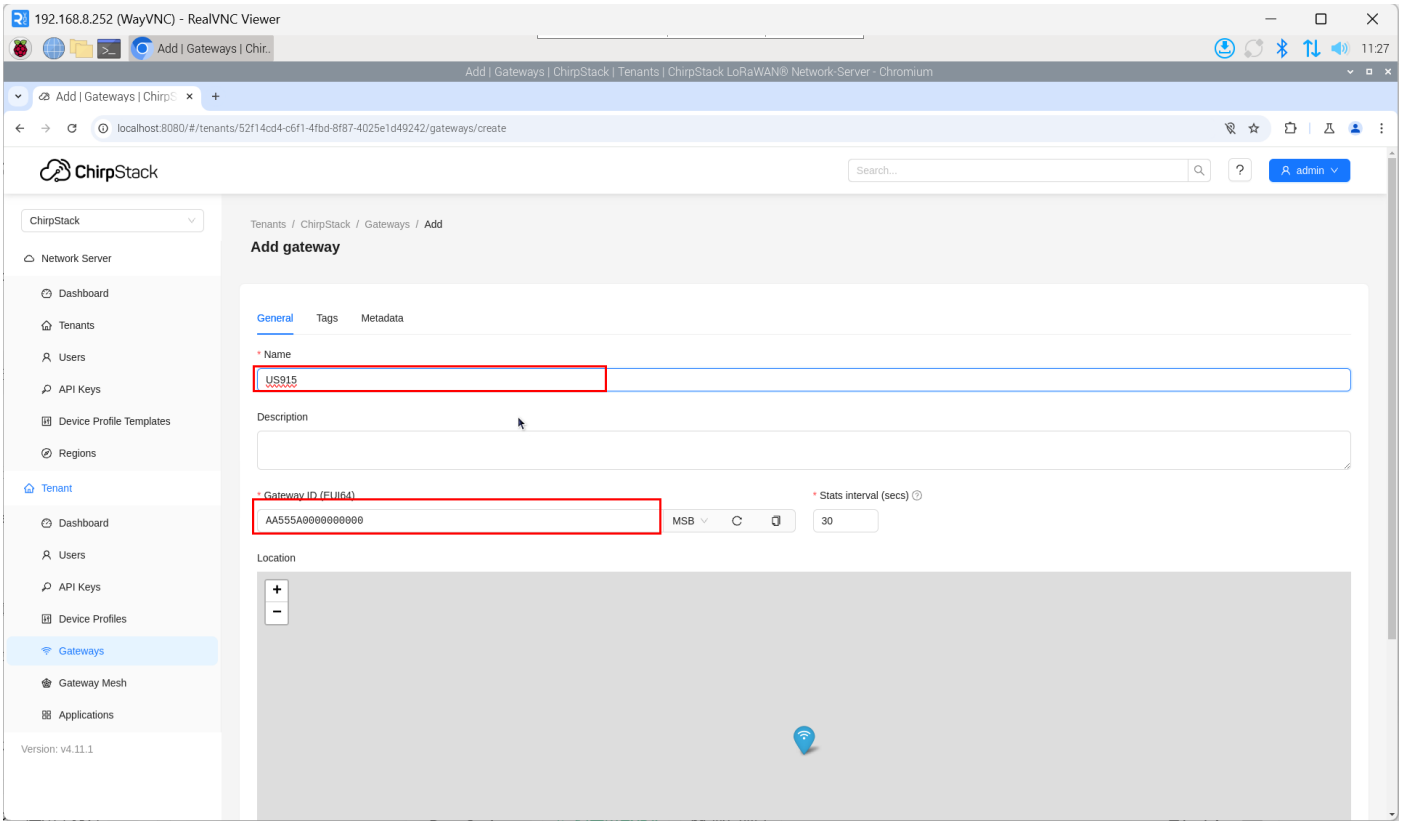
1. Open the ChirpStack management interface in a PC browser, then click "Gateway" → "Add gateway".

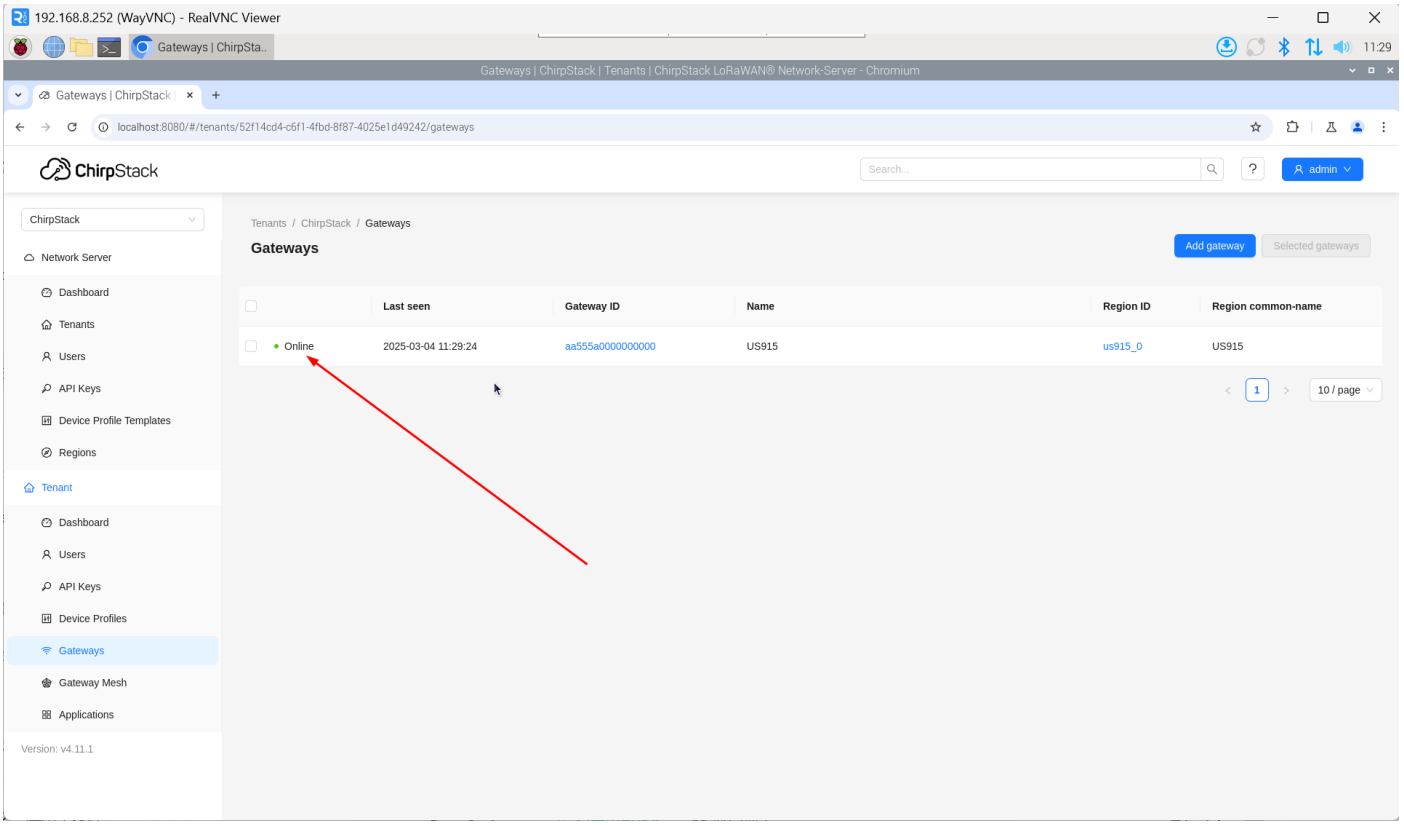


2. Click "add".



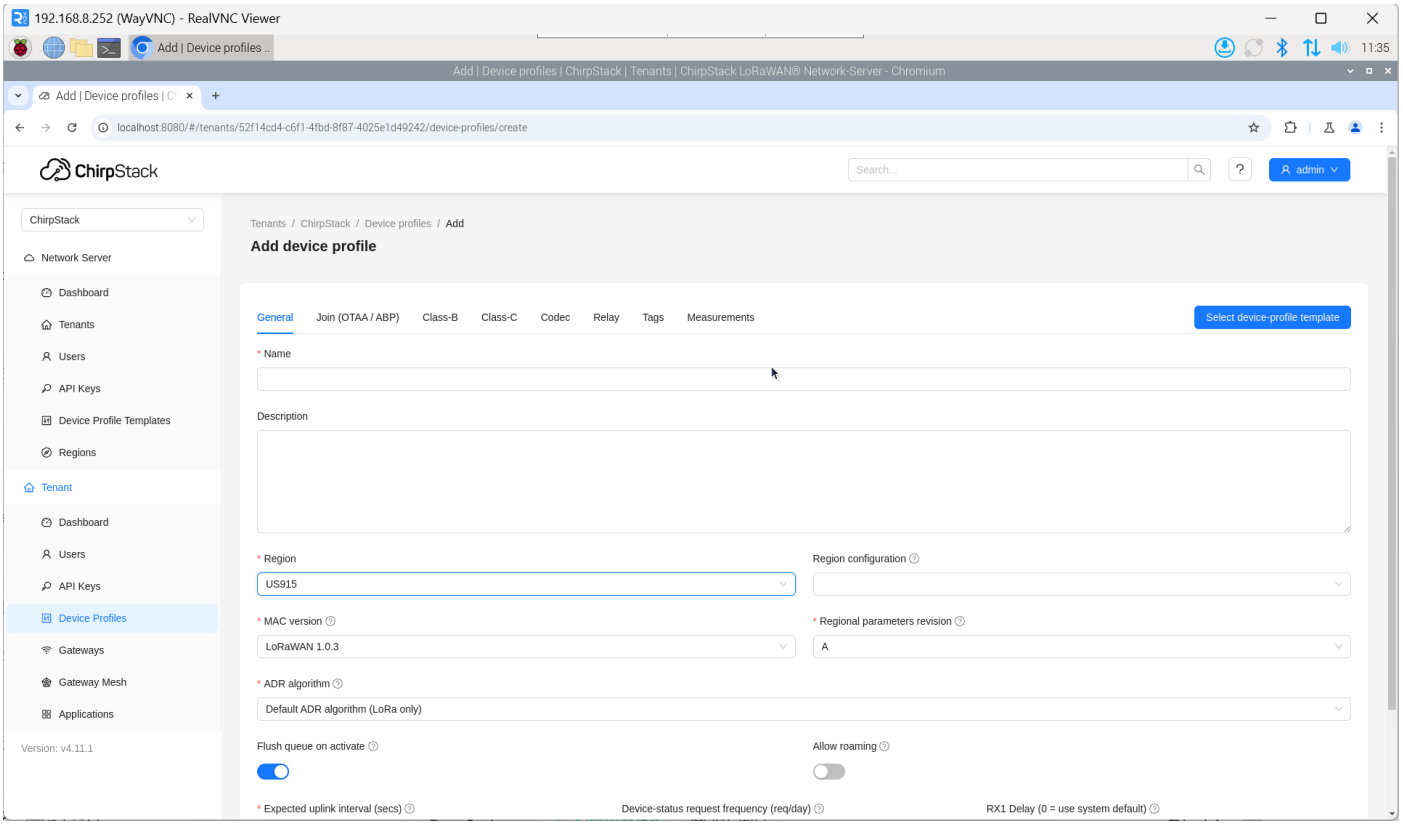
3. Enter the corresponding Gateway ID for your device, set the Name, then click "Submit". If the network connection is properly configured, the added gateway will transition to an Online status momentarily.





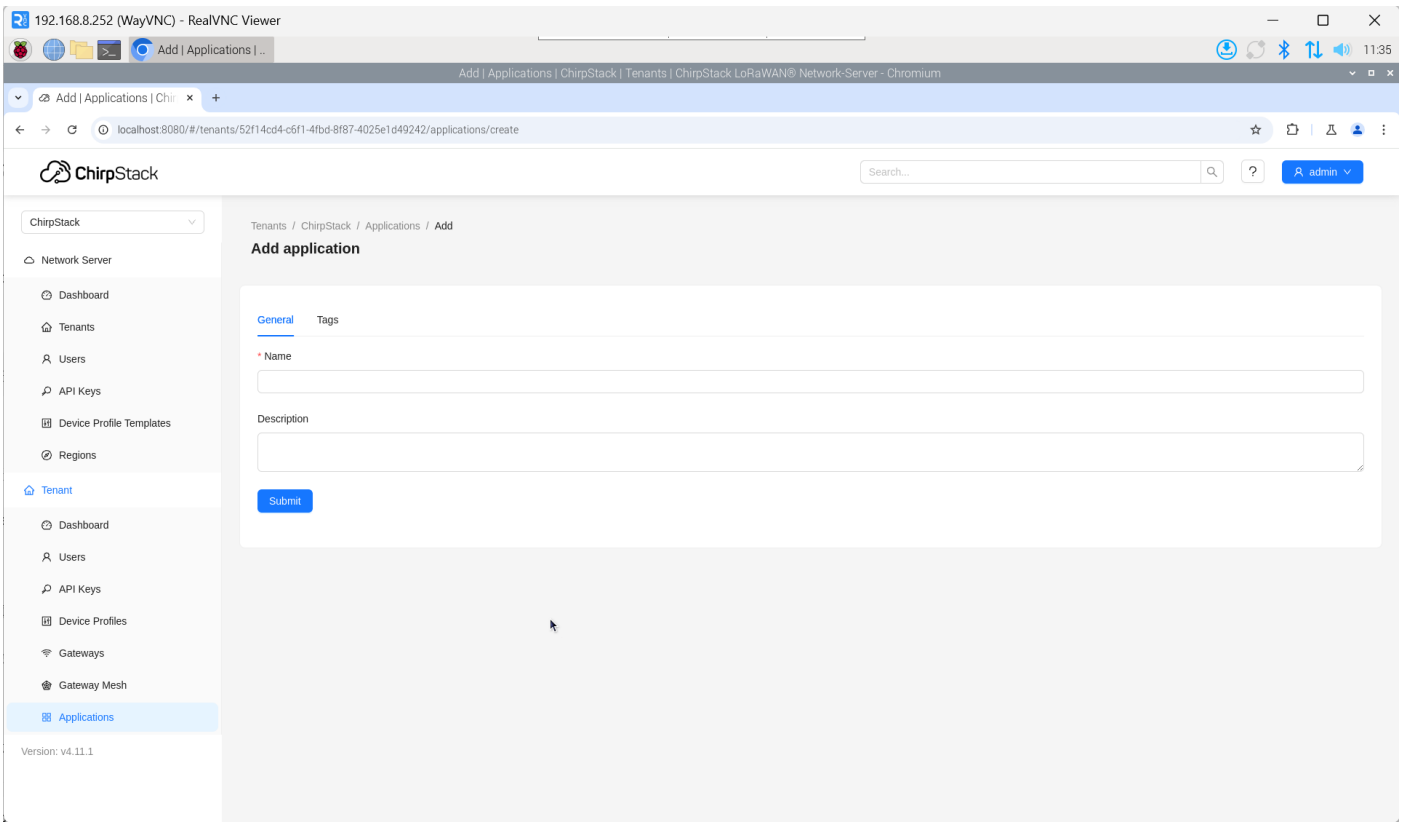
3.10.5.4 Add Device Profile

Click "Device Profile" → "Add device profile" to supplement device information.



3.10.5.5 Add Application

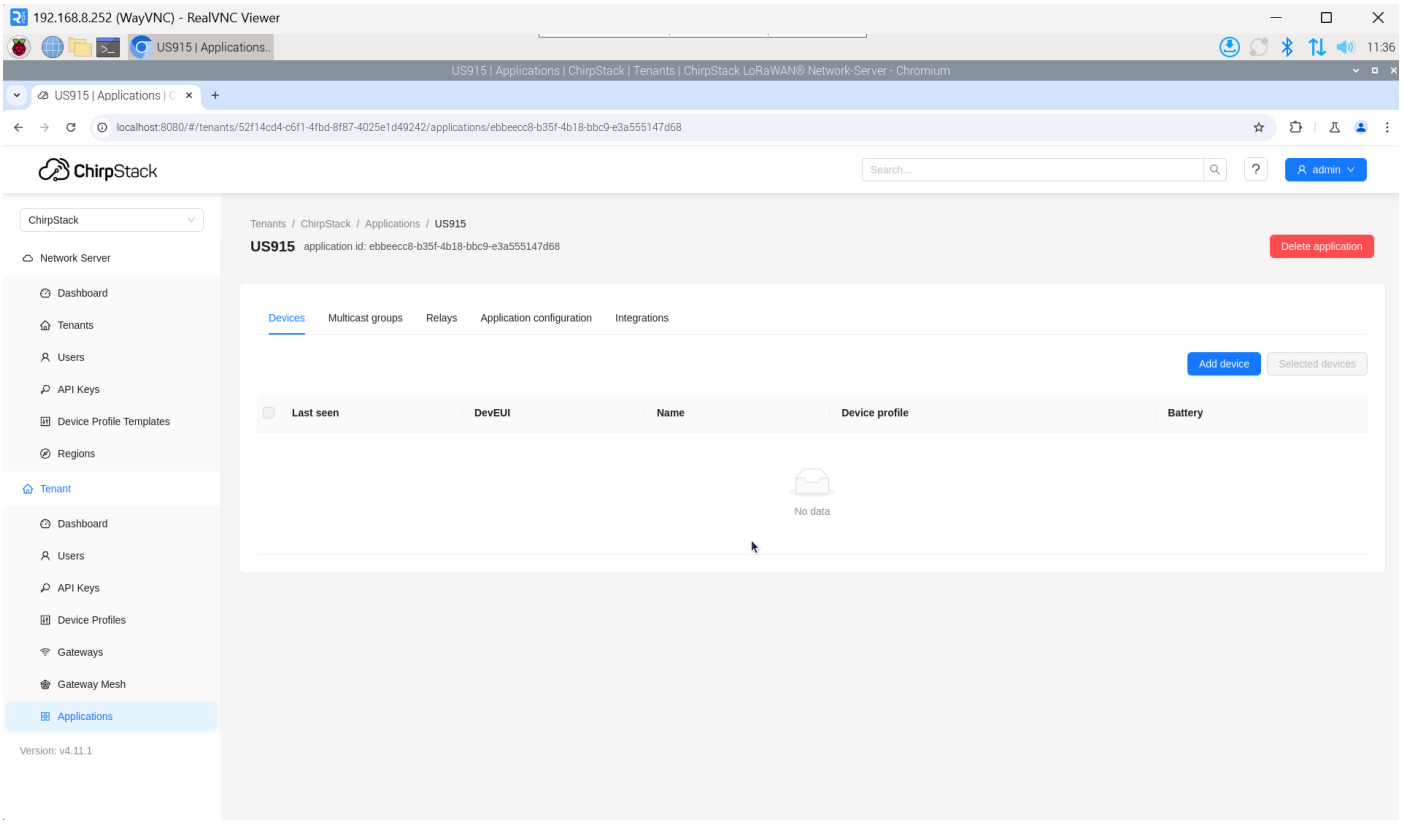
Click "Applications" → "Add application".

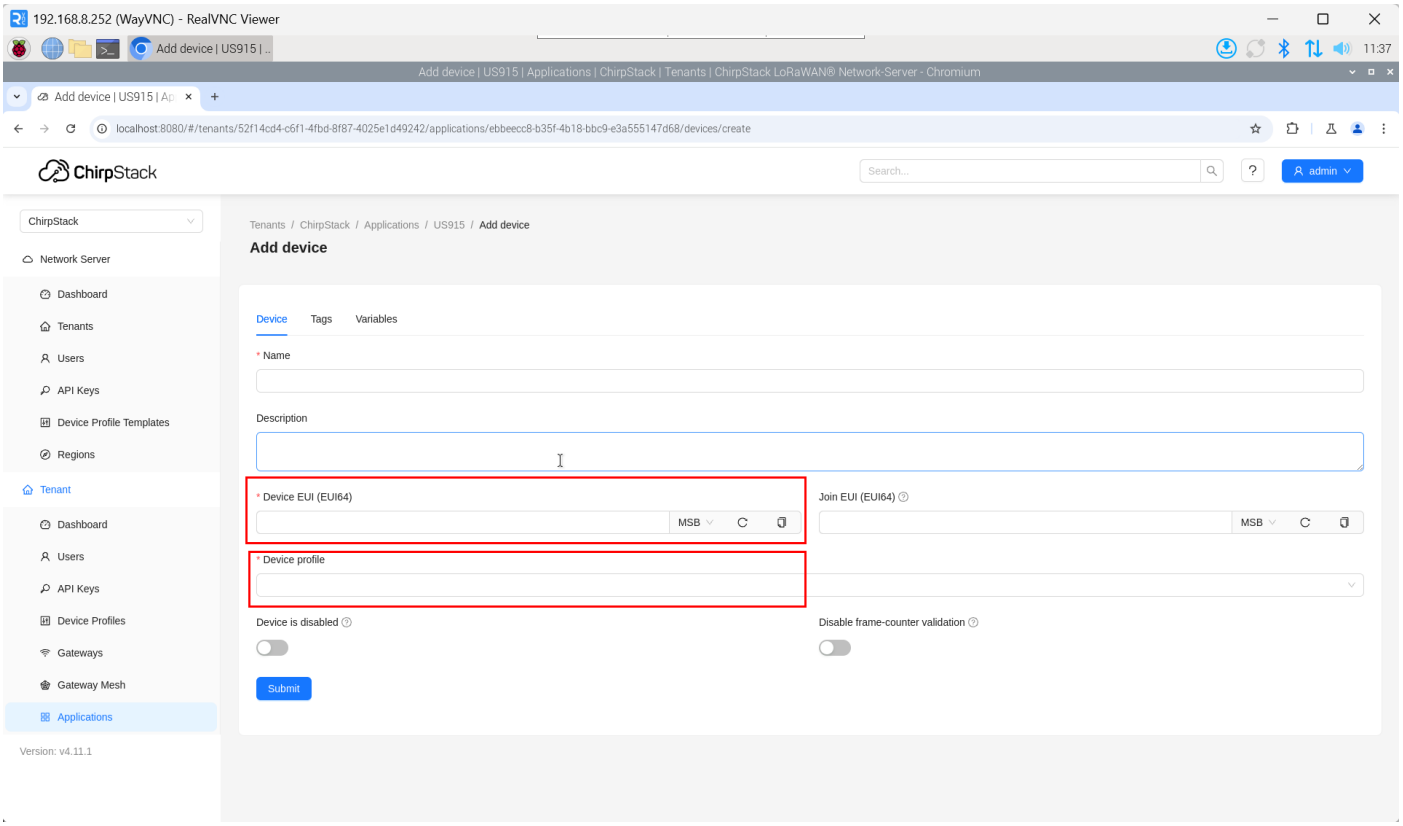


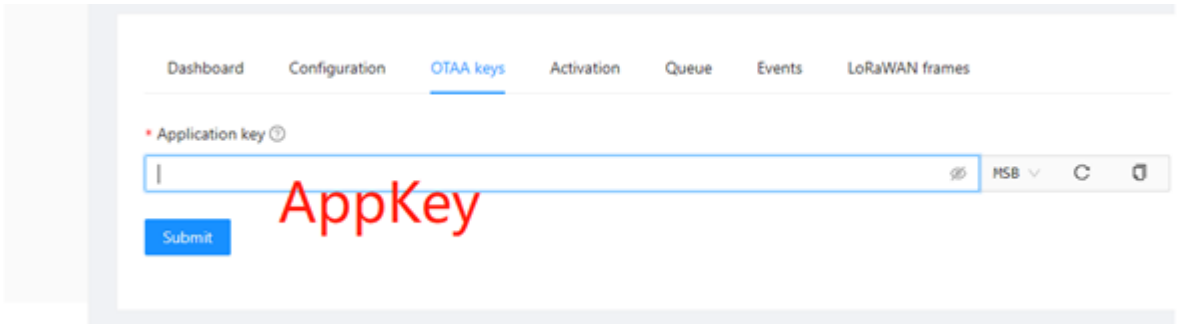
3.10.5.6 Add Device

The DevEUI and AppKey for LoRa end-device products are both provided by the LoRa device manufacturer.

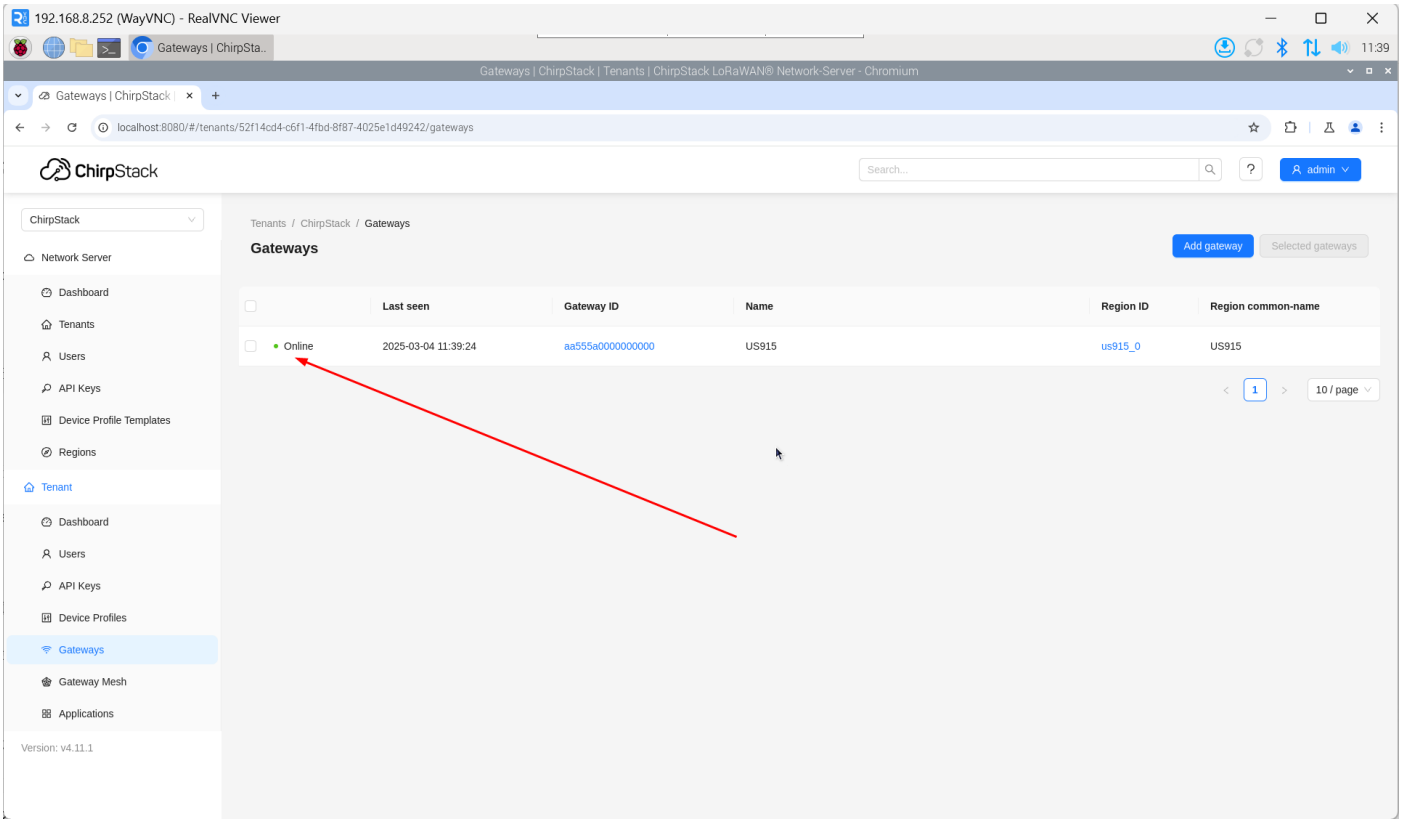
1. Click "Application" → "your application" → "Add device" to add the LoRa end-device.


192.168.8.252 (WayVNC) - RealVNC Viewer
US915 | Applications | ChirpStack | Tenants | ChirpStack LoRaWAN® Network-Server - Chromium
localhost:8080/#/tenants/52f14cd4-c6f1-4fbd-8f87-4025e1d49242/applications/ebbeecc8-b35f-4b18-bbc9-e3a555147d68
ChirpStack
admin
Tenants / ChirpStack / Applications / US915
US915 application id: ebbeecc8-b35f-4b18-bbc9-e3a555147d68
Delete application
Devices
Multicast groups
Relays
Application configuration
Integrations
Add device
Selected devices
Last seen
DevEUI
Name
Device profile
Battery
No data
Network Server
Dashboard
Tenants
Users
API Keys
Device Profile Templates
Regions
Tenant
Device Profiles
Gateways
Gateway Mesh
Applications
Version: v4.11.1


192.168.8.252 (WayVNC) - RealVNC Viewer
Add device | US915 | Applications | ChirpStack | Tenants | ChirpStack LoRaWAN® Network-Server - Chromium
localhost:8080/#/tenants/52f14cd4-c6f1-4fbd-8f87-4025e1d49242/applications/ebbeecc8-b35f-4b18-bbc9-e3a555147d68/devices/create
ChirpStack
admin
Tenants / ChirpStack / Applications / US915 / Add device
Add device
Device
Tags
Variables
* Name
Description
* Device EUI (EUI64)
MSB
Join EUI (EUI64)
* Device profile
Device is disabled
Disable frame-counter validation
Submit
Network Server
Dashboard
Tenants
Users
API Keys
Device Profile Templates
Regions
Tenant
Device Profiles
Gateways
Gateway Mesh
Applications
Version: v4.11.1


Dashboard
Configuration
OTAA keys
Activation
Queue
Events
LoRaWAN frames
* Application key
MSB
AppKey
Submit

2. After a few minutes, the device will appear in online status.



3.11 Encryption chip

ED-GWL2110 is equipped with ATECC608 encryption chip, which is connected to i2c-1 bus, and the default address of the device is 0x60.

atecc: https://github.com/wirenboard/atecc-util

```sh
atecc -b 1 -c 'serial'
```

4 Installing OS (optional)

The device is shipped with an operating system by default. If the OS is corrupted during use or the user needs to replace the OS, it is necessary to re-download the appropriate system image and install it. Our company supports to install the OS by installing the standard Raspberry Pi OS first, and then install the firmware package.

The following section describes the specific operations of image download, flashing to SD card and installation of firmware packages.

4.1 Downloading OS File

You can download the corresponding official Raspberry Pi OS file according to your actual needs, the download path is listed below:

OS	Download Path
Raspberry Pi OS(Desktop) 64-bit-bookworm (Debian 12)	https://downloads.raspberrypi.com/raspios_arm64/images/raspios_arm64-2024-07-04/2024-07-04-raspios-bookworm-arm64.img.xz (https://downloads.raspberrypi.com/raspios_arm64/images/raspios_arm64-2024-07-04/2024-07-04-raspios-bookworm-arm64.img.xz)
Raspberry Pi OS(Lite) 64-bit-bookworm (Debian 12)	https://downloads.raspberrypi.com/raspios_lite_arm64/images/raspios_lite_arm64-2024-07-04/2024-07-04-raspios-bookworm-arm64-lite.img.xz (https://downloads.raspberrypi.com/raspios_lite_arm64/images/raspios_lite_arm64-2024-07-04/2024-07-04-raspios-bookworm-arm64-lite.img.xz)
Raspberry Pi OS(Desktop) 32-bit-bookworm (Debian 12)	https://downloads.raspberrypi.com/raspios_armhf/images/raspios_armhf-2024-07-04/2024-07-04-raspios-bookworm-armhf.img.xz (https://downloads.raspberrypi.com/raspios_armhf/images/raspios_armhf-2024-07-04/2024-07-04-raspios-bookworm-armhf.img.xz)
Raspberry Pi OS(Lite) 32-bit-bookworm (Debian 12)	https://downloads.raspberrypi.com/raspios_lite_armhf/images/raspios_lite_armhf-2024-07-04/2024-07-04-raspios-bookworm-armhf-lite.img.xz (https://downloads.raspberrypi.com/raspios_lite_armhf/images/raspios_lite_armhf-2024-07-04/2024-07-04-raspios-bookworm-armhf-lite.img.xz)

4.2 Flashing to SD Card

The ED-GWL2110 supports booting the system from an SD card. Depending on your application needs, refer to the instructions below to flash the OS image.

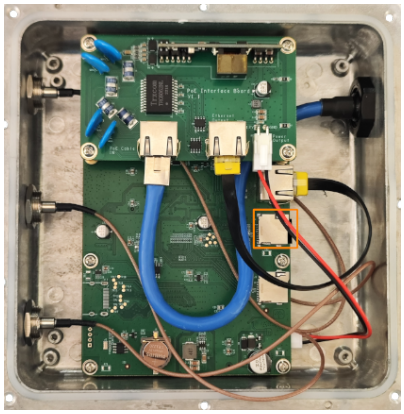
It is recommended to use the Raspberry Pi official tools. The download paths are as follows:

- Raspberry Pi Imager: https://downloads.raspberrypi.org/imager/imager_latest.exe (https://downloads.raspberrypi.org/imager/imager_latest.exe)
- SD Card Formatter: https://www.sdcardformatter.com/download/ (https://www.sdcardformatter.com/download/)

Preparation:

- The downloading and installation of the official tools to the computer have been completed.
- Opening the device enclosure and removing the Micro SD card:
  1. Unscrew the 9 screws on the bottom of the ED-GWL2110 counterclockwise using a screwdriver, as illustrated in the figure below.

  2. Remove the Micro SD card from the marked location shown in the figure below.

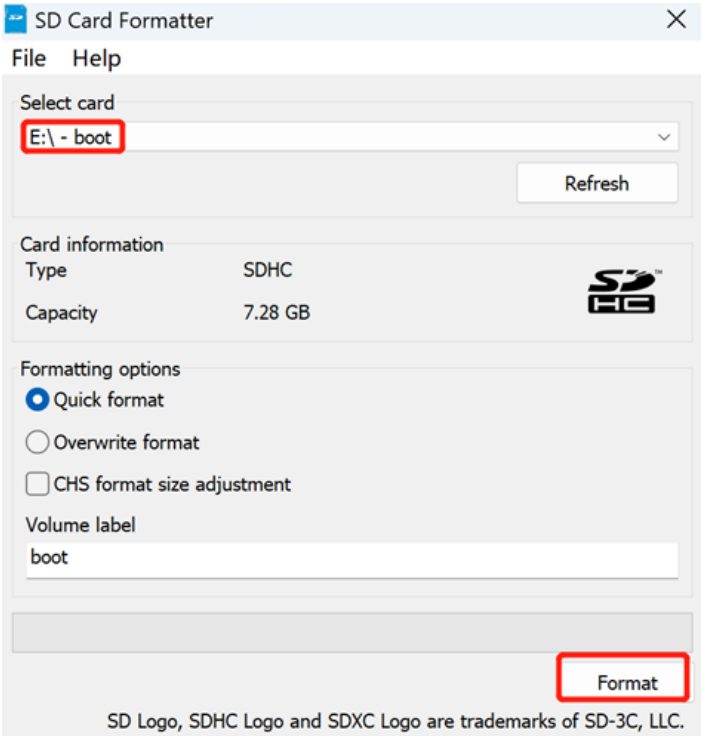


- The OS file has been obtained.
- An SD card reader has been prepared.
- Power has been disconnected.

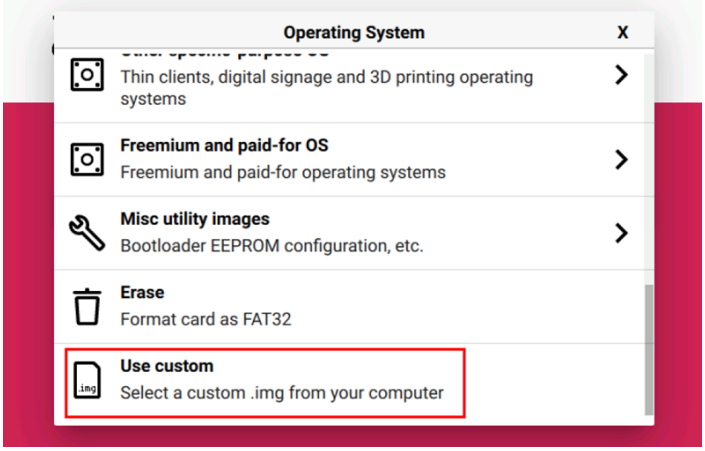
Steps:

The steps are described using Windows system as an example.

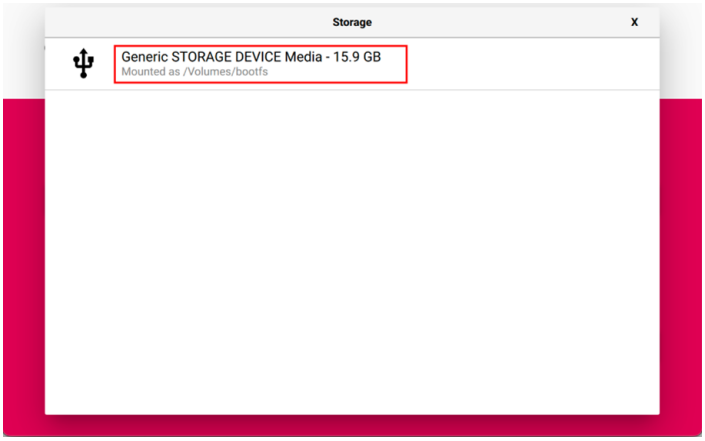
1. Insert the removed SD card into the card reader and then into the USB port of your computer.
2. Open SD Card Formatter, select the formatted drive letter, and click "Format" at the lower right to format.



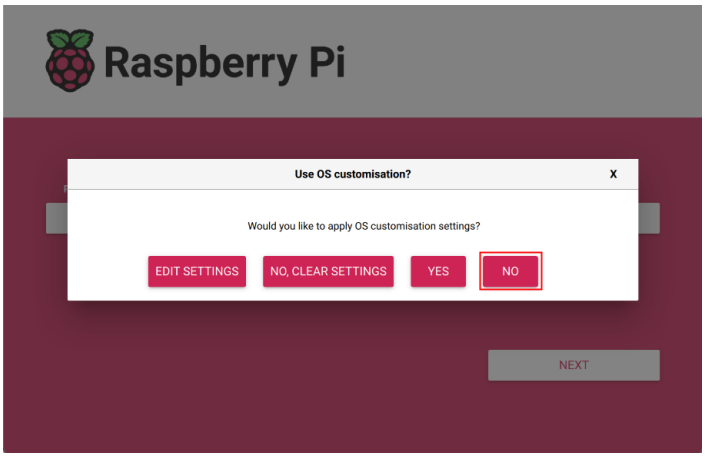
3. In the pop-up prompt box, select "Yes".
4. When the formatting is completed, click "OK" in the prompt box.
5. Close SD Card Formatter.
6. Open Raspberry Pi Imager, select "CHOOSE OS" and select "Use Custom " in the pop-up pane.



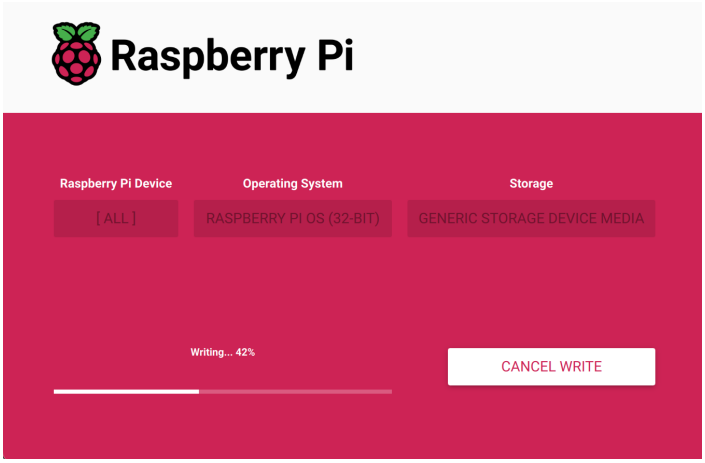
7. According to the prompt, select the OS file under the user-defined path and return to the main page.
8. Click "CHOOSE STORAGE", select the default device in the "Storage" interface, and return to the main page.



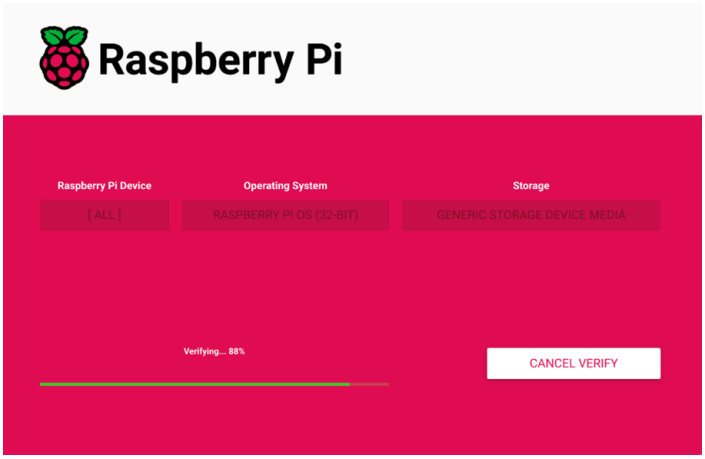
9. Click “NEXT”, select "NO " in the pop-up “Use OS customization?” pane.



10. Select “YES” in the pop-up “Warning” pane to start writing the image.



11. After the OS writing is completed, the file will be verified.



12. After the verification is completed, click “CONTINUE” in the pop-up “Write Successful” box.
13. Close the Raspberry Pi Imager, remove the card reader and SD card, and reinsert them into the device.

4.3 Installing Firmware Package

After flashing the standard Raspberry Pi OS onto the ED-GWL2110 series, you must configure the system by adding the edatec APT repository and installing the firmware package to ensure full functionality.

For the ED-GWL2110, three different LoRaWAN protocol frequency bands are available:

- 470MHz (CN470): China
- 868MHz (EU868): Europe
- 915MHz (US915): United States

LoRa Type	Firmware
470MHz(CN470)	gwl2110_470
868MHz(EU868)	gwl2110_868
915MHz(US915)	gwl2110_915

Preparation:

- The flashing to SD card of the Raspberry Pi standard OS (bookworm) has been completed.
- The device has booted normally and the relevant boot configuration has been completed.

Steps:

1. After the device starts normally, execute the following commands in the command pane to add the edatec apt source and installing firmware package.

- 470MHz(CN470)

```sh
curl -s https://apt.edatec.cn/bsp/ed-install.sh | sudo bash -s gwl2110_470
```

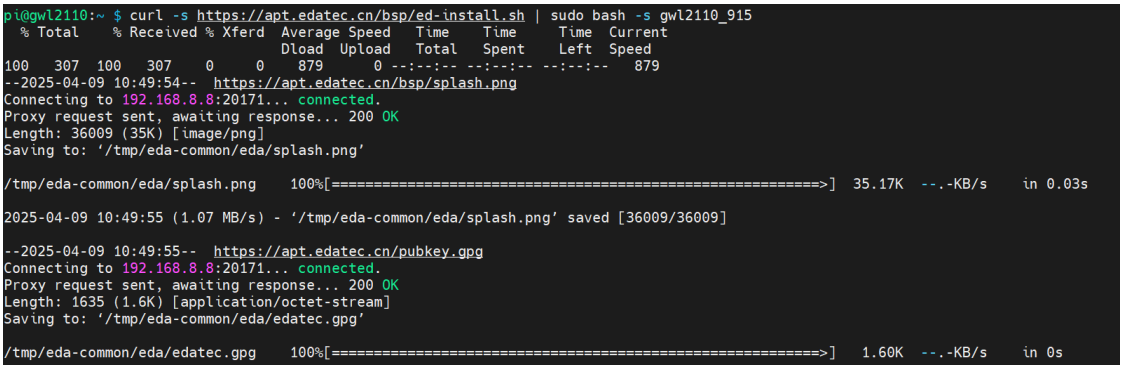
- 868MHz(EU868)

```sh
curl -s https://apt.edatec.cn/bsp/ed-install.sh | sudo bash -s gwl2110_868
```

- 915MHz(US915)

```sh
curl -s https://apt.edatec.cn/bsp/ed-install.sh | sudo bash -s gwl2110_915
```

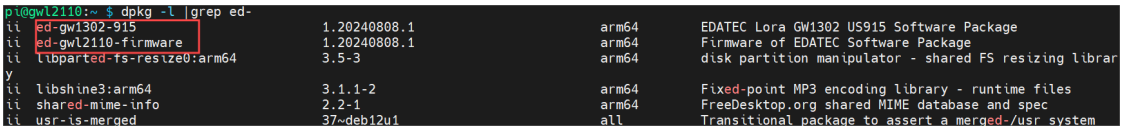
The figure below uses 915MHz (US915) as an example.



2. After the installation is complete, the system automatically reboots.
3. Execute the following command to check whether the firmware package is installed successfully.

```sh
dpkg -l | grep ed-
```

The result in the picture below indicates that the firmware package has been installed successfully.



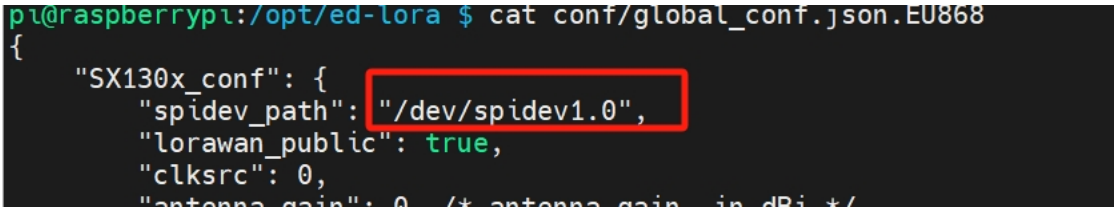
> **TIP**
>
> If you have installed the wrong firmware package, you can execute `sudo apt-get --purge remove package` to delete it, where “package” is the package name.

4. Execute the following command to enable the i2c interface.

```sh
sudo raspi-config nonint do_i2c 0
```

5. Execute the following commands in sequence to check the SPI configuration of LoRa. Then you need set `"the spidev_path"` to `"/dev/spidev1.0"`.

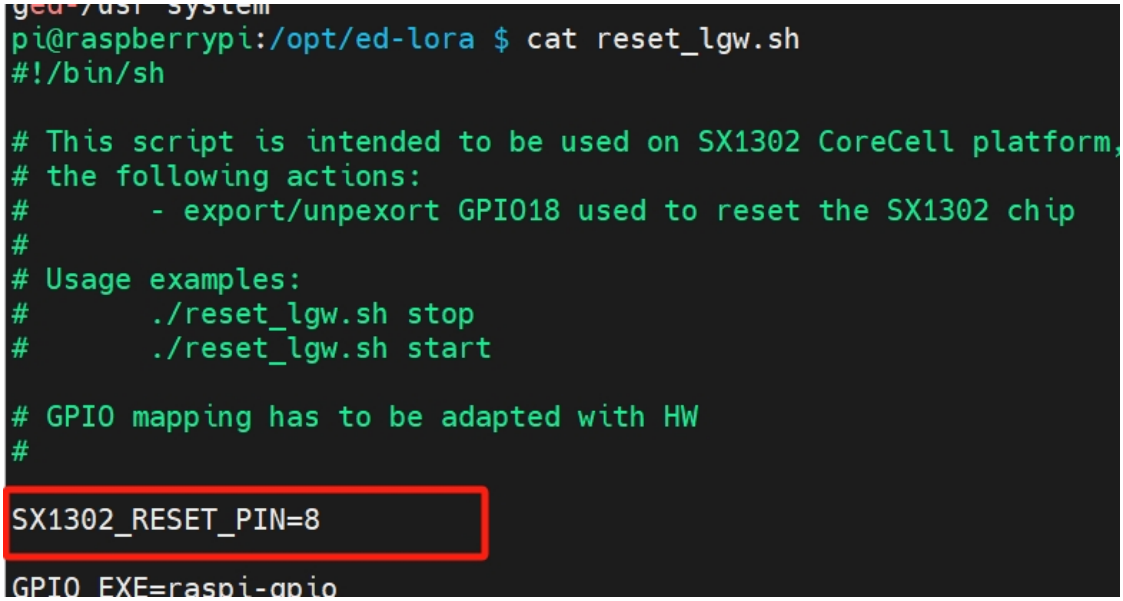
```sh
cd /opt/ed-lora
cat conf/global_conf.json.EU868
```



> TIP
>
> For US915/CN470 modules, replace `EU868` with `US915` or `CN470`.

6. Execute the following commands in sequence to check the reset pin of LoRa. Then you need set the reset pin to `8`.

```sh
cd /opt/ed-lora
cat reset_lgw.sh
```

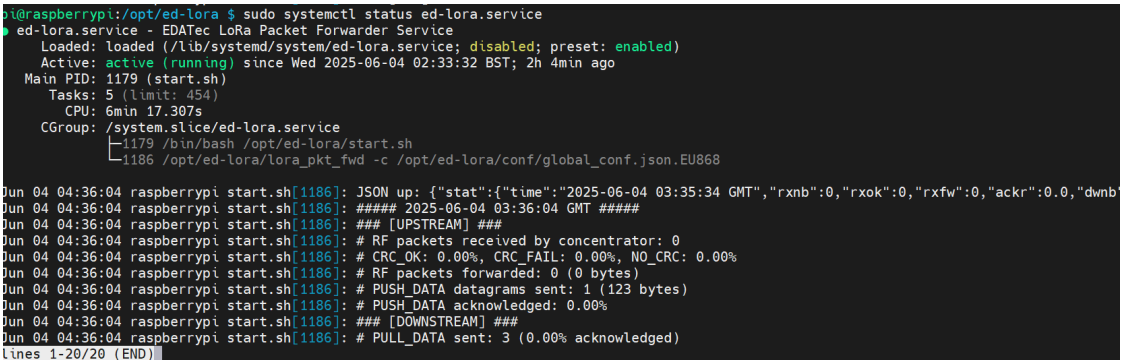


7. Execute the following command to restart the device.

```sh
sudo reboot
```

8. Execute the following command to to check the status of the LoRa service.

```sh
sudo systemctl status ed-lora.service
```



- If the status of LoRa service is running, it's OK.
- If the status of LoRa service is not running, please Execute `sudo systemctl start ed-lora.service` to start it manually.