

# **LoRaWAN Stack for VAS sensors**

- Overview of LoRaWAN Stack for VAS sensors -

Document version:	1.2
Last modified:	20/05/2024 10:24:00

# Contents

<b>1. Overview of the LoRaWAN Stack .....</b>	<b>2</b>
1.1 GNU/Linux server.....	2
1.2 LoRaWAN Gateway .....	2
1.3 LoRaWAN Network Server .....	3
1.4 IoT Platform .....	3
1.5 Reverse Proxy .....	3
1.6 Mandatory configuration and requirements .....	3
<b>2. Configuration of the LoRaWAN Network Server (ChirpStack) .....</b>	<b>4</b>
2.1 Example of Docker installation on Debian 12 .....	4
2.2 Install Docker Compose .....	5
2.3 Clone LoRaWAN Stack for VAS applications example repository and start it ....	5
2.4 Add LoRaWAN Gateway to the ChirpStack .....	7
2.5 Create certificates for the LoRaWAN gateway.....	7
2.6 Add new VAS devices .....	9
2.7 Connect ChirpStack device to the ThingsBoard .....	12
<b>3. Configuration of the LoRaWAN Gateway (RAK7289CV2 WisGate Edge Pro)...</b>	<b>16</b>
3.1 Initial configuration.....	16
3.2 Configure LoRa settings.....	19
3.3 Edit hosts file on the gateway .....	20
<b>References: .....</b>	<b>22</b>

## Revision History

Version 0.1

Initial version: 25 April 2024

Version 1.0

Added Configuration and Usage Guide: 14 May 2024

Version 1.1

Updated Configuration and Usage Guide: 15 May 2024

Version 1.2

Updated Configuration and Usage Guide: 17 May 2024

# 1

## Overview of the LoRaWAN Stack

A stack was implemented to connect the VAS sensors over the LoRaWAN network, collect data from these sensors, and visualise it. The main components of the stack are:

- GNU/Linux server (Debian);
- LoRaWAN Gateway (RAK7289CV2 WisGate Edge Pro);
- LoRaWAN Network Server (ChirpStack);
- IoT Platform (ThingsBoard);
- Reverse Proxy (Traefik).

### 1.1 GNU/Linux server

All software components are installed on a Debian GNU/Linux 12 (Bookworm) server. The server is running with the following hardware configuration:

- 4 CPU cores;
- 8 GB RAM;
- 2 Hard disks (32 GB for system disk and 200 GB disk for data).
  - NOTE: 2<sup>nd</sup> HDD is optional, everything can be stored on one hard drive.

All ChirpStack components, ThingsBoard, and Traefik are installed on Docker, running in rootless mode using Docker Compose.

The following firewall ports are opened:

- 22 for SSH;
- 443 for HTTPS;
- 8883 for MQTT.

The following firewall ports can be opened for debugging:

- 8080 for ChirpStack Web GUI without TLS;
- 9090 for ThingsBoard Web GUI without TLS.

### 1.2 LoRaWAN Gateway

RAK7289CV2 WisGate Edge Pro is used as a LoRaWAN Gateway [1]. The following hardware configuration is employed:

- 8 Channels;
- No LTE;
- EU868 Frequency Region.

The gateway communicates with ChirpStack using an MQTT v3.1 Bridge with TLS encryption on port 8883.

## 1.3 LoRaWAN Network Server

ChirpStack v4 is used as the LoRaWAN Network Server [2]. All of its components (chirpstack, chirpstack-rest-api, postgres, redis, mosquitto) are installed on Docker, which is running in rootless mode using Docker Compose. Only the EU region is enabled for LoRaWAN communication.

The LoRaWAN device repository has been imported into ChirpStack.

An MQTT connection with TLS encryption is enabled on port 8883. Port 1883 for external connections without encryption is disabled, although it can be used internally within the mosquitto container for testing and debugging.

## 1.4 IoT Platform

The ThingsBoard Community Edition is installed as an IoT platform for data collection, processing, and visualisation [3]. An instance with a Cassandra database and Kafka queue service is used, as recommended by ThingsBoard developers for a production environment [4]. 8 GB of RAM is recommended for this setup.

## 1.5 Reverse Proxy

Out of the box, ChirpStack and ThingsBoard do not have HTTPS capabilities – they only support unencrypted connections to access their Web Dashboards. Therefore, Traefik is used as a reverse proxy to implement connections with TLS encryption [5].

Self-signed certificates are utilized, however, there is also the option to use Let's Encrypt certificates with valid domain names.

The following domains are used:

- ChirpStack Dashboard: <https://chirpstack.vas.internal/>
- ThingsBoard Dashboard: <https://thingsboard.vas.internal/>
- Traefik Dashboard: <https://traefik.vas.internal/>

The Traefik Dashboard is password-protected using the basicAuth middleware from Traefik.

## 1.6 Mandatory configuration and requirements

- The LoRaWAN Gateway and ChirpStack must be on the same network.
- Static IP addresses are required.
- The correct IP address of the ChirpStack must be set on the LoRaWAN Gateway.
- Domains used for the reverse proxy must be configured on the DNS server, which is not included in this setup.

# 2

## Configuration of the LoRaWAN Network Server (ChirpStack)

Docker compose is used to run LoRaWAN Network Server (ChirpStack), therefore docker and docker compose should be installed on the system.

Installation instructions for various operating systems can be found at the following page:

<https://docs.docker.com/engine/install/>

Installation example on Debian 12 using the apt repository is presented in chapter 2.1.

Also, as mentioned in the Chapter 1, ports 22, 443, 8883, 8080, 9090 should be open.

### 2.1 Example of Docker installation on Debian 12

Uninstall old versions, if installed:

```
for pkg in docker.io docker-doc docker-compose podman-docker containerd
runc; do sudo apt-get remove $pkg; done
```

Update apt package index

```
apt update
```

Install packages to allow apt to use a repository over HTTPS:

```
apt install apt-transport-https ca-certificates curl gnupg lsb-release
software-properties-common
```

Set up Docker's apt repository.

```
# Add Docker's official GPG key:
```

```
sudo apt-get update
```

```
sudo apt-get install ca-certificates curl
```

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
sudo curl -fsSL https://download.docker.com/linux/debian/gpg -o
/etc/apt/keyrings/docker.asc
```

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
# Add the repository to Apt sources:
```

```
echo \
"deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/debian \
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
sudo apt-get update
```

Install latest version

```
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-
plugin docker-compose-plugin
```

Verify that the installation is successful by running the hello-world image:

```
docker run hello-world
```

## 2.2 Install Docker Compose

Install the Compose plugin

```
sudo apt update  
sudo apt install docker-compose-plugin
```

Install Compose standalone.

```
sudo curl -SL  
https://github.com/docker/compose/releases/download/v2.24.6/docker-  
compose-linux-x86_64 -o /usr/local/bin/docker-compose  
sudo chmod +x /usr/local/bin/docker-compose
```

Note that Compose standalone uses the `-compose` syntax instead of the current standard syntax `compose`.

For example type `docker-compose up` when using Compose standalone, instead of `docker compose up`.

## 2.3 Clone LoRaWAN Stack for VAS applications example repository and start it

Configuration example of LoRaWAN Stack for VAS applications is provided in the following GitHub repository:

[https://github.com/NEUROTECHLT/LoRa\\_WAN-Server-Deploy.git](https://github.com/NEUROTECHLT/LoRa_WAN-Server-Deploy.git)

Install git (if it is not already installed:

```
sudo apt update && sudo apt install git
```

Install git lfs

```
curl -s https://packagecloud.io/install/repositories/github/git-  
lfs/script.deb.sh | sudo bash  
sudo apt-get install git-lfs
```

Clone the repository

```
cd ~  
git clone https://github.com/NEUROTECHLT/LoRa\_WAN-Server-Deploy.git
```

Fetch LFS files:

```
cd ~/LoRa_WAN-Server-Deploy  
git lfs pull
```

Delete all `.gitkeep` files, which were used to force git to track empty folders:

```
cd ~/LoRa_WAN-Server-Deploy  
sudo find . -type f -name ".gitkeep" -exec rm -f {} +
```

Fix permissions of the ThingsBoard folders:

```
cd ~/LoRa_WAN-Server-Deploy  
sudo chown -R 799:799 configuration/thingsboard-data  
sudo chown -R 799:799 configuration/thingsboard-logs  
sudo chmod 0700 configuration/thingsboard-data/db
```

Start LoRaWAN Stack for VAS applications:

```
cd ~/LoRa_WAN-Server-Deploy  
sudo docker-compose up
```

OR:

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

## LoRaWAN Stack for VAS sensors

To stop LoRaWAN Stack for VAS applications:

```
cd LoRa_WAN-Server-Deploy
sudo docker-compose stop
```

OR:

```
sudo docker-compose down
```

NOTE: last command will discard the containers and the networks they were utilizing

All services should be accessible:

- ChirpStack Dashboard: <https://chirpstack.vas.internal/>
  - OR: `http://<IP_OF_THE_SERVER>:8080`
- ThingsBoard Dashboard: <https://thingsboard.vas.internal/>
  - OR: `http://<IP_OF_THE_SERVER>:9090`
- Traefik Dashboard: <https://traefik.vas.internal/>

As mentioned in Chapter 1, **all used \*.vas.internal domain names should be resolvable to the IP address of the server**. System administrator must ensure that by editing settings of the used DNS server or host files or any other means. E.g. following steps can be used to edit hosts file on Linux if these domain names should be resolved to localhost (127.0.0.1):

Open /etc/hosts file in nano text editor:

```
sudo nano /etc/hosts
```

Append to the end of the file:

```
127.0.0.1      chirpstack.vas.internal      thingsboard.vas.internal
traefik.vas.internal
```

Following default usernames and passwords are used:

- ChirpStack:
  - User: admin
  - Password: LoRaChirpStack2)@\$
- ThingsBoard:
  - System Administrator:
    - User: sysadmin@thingsboard.org
    - Password: sysadmin
  - Tenant Administrator:
    - User: tenant @thingsboard.org
    - Password: tenant
  - Customer User:
    - User: customer @thingsboard.org
    - Password: customer
- Traefik Dashboard:
  - User: vasuser
  - Password: LoRaChirpStack2)@\$

NOTE:

Docker automatically resolves addresses of the services defined in the docker-compose file. But sometimes it does not work e.g. chirpstack can't resolve postgres address. As a workaround these names can be added to the hosts file of the server. E.g.:

Edit hosts file

```
sudo nano /etc/hosts
```

Append to the hosts file:

```
127.0.0.1 chirpstack mosquitto postgres
```



## 2.4 Add LoRaWAN Gateway to the ChirpStack

Open Web GUI of the ChirpStack by loading following web page:

<https://chirpstack.vas.internal/>

OR:

http://<IP\_OF\_THE\_SERVER>:8080/

Navigate to: Tenant → Gateways

Press “Add gateway” button at the top right corner of the page.

Fill mandatory fields as shown in Figure 1:

- Name: can be any, e.g. WisGate\_Edge\_Pro\_RAK7289V2
- Description: can be any, e.g.: WisGate Edge Pro (RAK7289-V2)
- Gateway ID (EUI64): should match EUI of the used LoRaWAN gateway.
  - In the case of the RAK7289CV2 WisGate Edge Pro, it can be found in the Overview page of its Web GUI, as shown at the end of the Chapter 3.1.

Figure 1. Add LoRaWAN Gateway

## 2.5 Create certificates for the LoRaWAN gateway

Certificates are needed to allow MQTT connection with TLS encryption.

Open Web GUI of the ChirpStack by loading following web page:

<https://chirpstack.vas.internal/>

OR:

http://<IP\_OF\_THE\_SERVER>:8080/

Navigate to: Tenant → Gateways → <GATEWAY\_ADDED\_IN\_PREVIOUS\_STEP> → → TLS certificate

Press “Generate certificate”.

Following fields of the generated certificate should be shown (examples of these certificates are shown in Figure 2 and Figure 3):

- CA certificate
  - Copy everything, including “-----BEGIN CERTIFICATE-----” and “-----END CERTIFICATE-----” lines to the text file and save it as **ca.pem** file.

## LoRaWAN Stack for VAS sensors

- TLS certificate
  - Save everything, including “-----BEGIN CERTIFICATE-----” and “-----END CERTIFICATE-----” lines to the text file and save it as **client\_cert.pem** file.
- TLS key
  - Save everything, including “-----BEGIN CERTIFICATE-----” and “-----END CERTIFICATE-----” lines to the text file and save it as **client\_key.pem** file.

These three certificate files will be needed while configuring LoRaWAN gateway in Chapter 3.2.

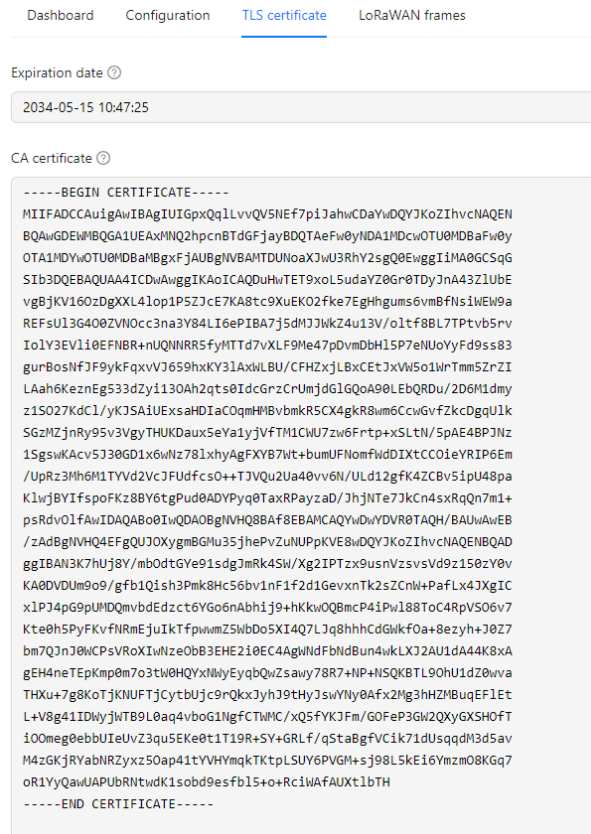


Figure 2. Example of the generated certificate for the gateway – CA certificate

TLS certificate ⓘ

```

-----BEGIN CERTIFICATE-----
MIIDUzCCATugAwIBAgIUdWDrndh+uWrdkIVX3r9m1mXyS1gwDQYJKoZIhvcNAQEL
BQAwGDEWMBQGA1UEAxMNQ2hpcnBtdGFjayBDQTAeFw0yNDA1MTQxMjM1MjVhFw0z
NDA1MTUwNzQ3MjVhMBsxGTAXBgNVBAMtZmU1UzZlNzQ0MGNkYmE4NDYyYmVhTATBgq
hkjOPQIBBgggqhkiJOPQIBBwNCAAAQZnN14qTKXdd7A2+T8KuHSTfIJHmoEFhtCpPF/
EAA11FXhaB2xOmRrYDo5qxNVC41Y0Sbe/AB4FV5AkR0/yq4to10wWzAJBgNVHRME
AjAAFA4GA1UdDwEB/wQEAwIHgDAdBgNVHQ4EFgQUAT3+5G6BkX0wK3IwByM7Vx9K
u1EwHwYDVR0jBBgwFoAUJ0XygmBGHu35jhePvZuNUPPkVE8wDQYJKoZIhvcNAQEL
BQADggIBANwqJ+I6jCgYybMdnT67Jzo0AG5HmJKXJKQ/dOUAdM+KLDcsyJmTID0g
LMUfwp1417Pb7jNL0+GvQyLYEetvbsp3+Mm4GoLNpwaPc9pf1Rq6LNLCMJ44pD3E
s0XQU5zyRD0NhVb1MKDoHPZE610csHVCYZnFzPc2tKMyFnBaFpM3FdCupKKnDn
yKaUw/3w7wLrDZoUvSUVuHQ+oGLEH00a+UYW6D0Tzn+ckJErS2AAwMjJmt4Pzhjr
8/gz9vr98UwhsUZ2rXTmoUoucFyv4PHa8wX0Yb00Pt3420jgeaY470Z6V+k1skTN
k5Qj+D07WCrFpmRvGGQ/yTAwvh0PIGxAud1vnomaGjmaCd+3/8AvISQ5Vm8GjY4e
1aOWYwNzQVvtFCTNb0shqkKjK1RhghWR/ge0MQakxEZXwX7vToecI8E13F6cjDidN0
oTZyPzzyw0HfcSTIRjVv0M84p51bd8L9DKkf/4SJoIiasPTjrSKJzxAYmQYUjaTp
twHfSd8jRL63Wzmp1vHqM3GqWnfQ0GygkxIsejgwAFBQLdUG0lwCm73OVmJ65FDg
K07HsP5R1pgI8s5LHPv++/dxUdHfIUsHqYdaFr4JkxYdDIHVJLsT3sPxtj0iFQF
IAV3pQIyEgjlTEvFrr2jGnKz14iFudoh2+Zkctd8g6RF/Eqp4q1A
-----END CERTIFICATE-----

```

TLS key ⓘ

```

-----BEGIN PRIVATE KEY-----
MIGHAgEAMBMGBYqGSM49AgEGCCqGSM49AwEHBG0wawIBAQQgNcheEQIKzLw0DTaM1
/+WszihdQE1QIT0B8gHZwAvUr4KhRANCAAAQZnN14qTKXdd7A2+T8KuHSTfIJHmo
EFhtCpPF/EAA11FXhaB2xOmRrYDo5qxNVC41Y0Sbe/AB4FV5AkR0/yq4t
-----END PRIVATE KEY-----

```

Figure 3. Example of the generated certificate for the gateway – TLS certificate and TLS key

## 2.6 Add new VAS devices

By default, VAS devices are programmed to use Over-The-Air-Activation (OTAA). Following steps can be used to add VAS device using OTAA.

Open Web GUI of the ChirpStack by loading following web page:

<https://chirpstack.vas.internal/>

OR:

http://<IP\_OF\_THE\_SERVER>:8080/

Navigate to: Tenant → Applications → VAS Application

Press “Add device” button as shown in the Figure 4.

## LoRaWAN Stack for VAS sensors

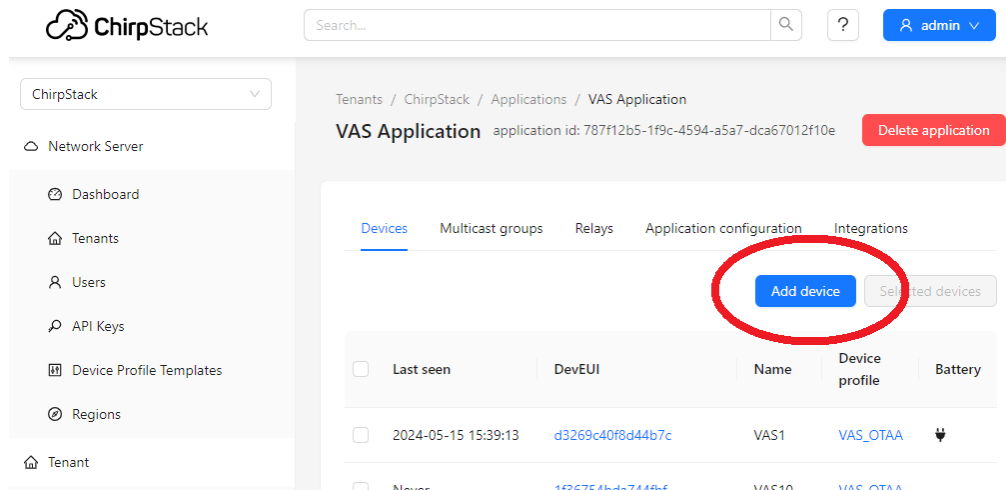


Figure 4. Add VAS device

Fill in mandatory fields, as shown in Figure 5:

- Name: could be any, e.g. VAS2
- Description: can be any, e.g. 2nd VAS device
- Device EUI (EUI64): Has to match DevEUI of the VAS device, e.g.:
  - 56bfa08c5b549e71
- Device profile: VAS\_OTAA

Press “Submit” button after entering this information.

After that, “OTAA keys” menu should open automatically.

Enter following default Application key as shown in Figure 6 and press “Submit”:

2b7e151628aed2a6abf7158809cf4f3c

After that, VAS device should connect and activity should be seen in the “Events” and “LoRaWAN frames” menu entries of the VAS device as shown in Figure 7 and Figure 8.

LoRaWAN Stack for VAS sensors

ChirpStack

Network Server

- Dashboard
- Tenants
- Users
- API Keys
- Device Profile Templates
- Regions

Tenant

- Dashboard
- Users
- API Keys
- Device Profiles
- Gateways

Add device

Device

Tags

Variables

\* Name

VAS2

Description

2nd VAS device

\* Device EUI (EUI64)

56bfa08c5b549e

MSB

Join EUI (EUI64)

MSB

\* Device profile

VAS\_OTAA

Device is disabled

Disable frame-counter validation

Submit

Figure 5. Mandatory parameters of the VAS device

Dashboard

Configuration

OTAA keys

Activation

Queue

Events

\* Application key

2b7e151628aed2a6abf7158809cf4f3c

MSB

Submit

Figure 6. Default OTAA Application key

Tenants / ChirpStack / Applications / VAS Application / Devices / VAS1

VAS1 device eui: d3269c40f8d44b7c

Delete device

Dashboard

Configuration

OTAA keys

Activation

Queue

Events

LoRaWAN frames

Download

2024-05-15 16:00:18	<div>up</div>	<div>DR: 5</div> <div>Data: 01190800070008007f005c0b2e0b9b0a360c2300220026005d02780078008a008f070000000000000020064016d01000073011d00</div> <div>FCnt: 19094</div> <div>FPort: 2</div>
2024-05-15 16:00:08	<div>up</div>	<div>DR: 5</div> <div>Data: 01190800070008007900720b370ce60b810c250024002a0093027d0074008a000b080000000000002005c016b01000093011d00</div> <div>FCnt: 19093</div> <div>FPort: 2</div>
2024-05-15 15:59:59	<div>up</div>	<div>DR: 5</div> <div>Data: 01190800070008007900720b370ce60b810c250024002a0093027d0074008a000b080000000000002005c016b01000093011d00</div> <div>FCnt: 19092</div> <div>FPort: 2</div>

Figure 7. Events of the VAS device

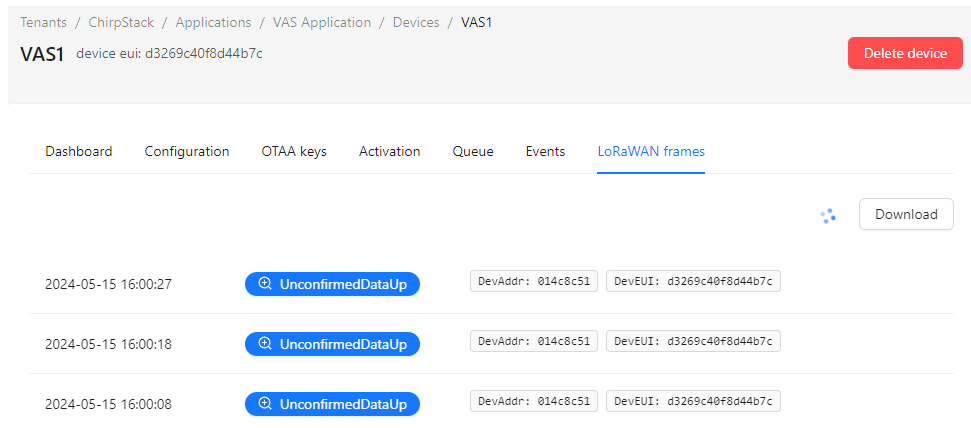


Figure 8. LoRaWAN frames of the VAS device

## 2.7 Connect ChirpStack device to the ThingsBoard

First, device has to be created in the ThingsBoard.

- Open Web GUI of the ThingsBoard by loading following web page:
  - <https://thingsboard.vas.internal/>
  - OR:
  - [http://<IP\\_OF\\_THE\\_SERVER>:9090/](http://<IP_OF_THE_SERVER>:9090/)
- Login with tenant account.
- Navigate to: Entities → Devices
- Press “+” icon in the top right corner and press Add new device as shown in Figure 9.
- “Add new device” menu should open, as shown in Figure 10.
- Enter name, e.g. VAS2, select “default” profile and press Add button.
- A window will open where you can check the device's connection to the ThingsBoard as shown in Figure 11. This step is not needed at the moment, therefore, simply close it.

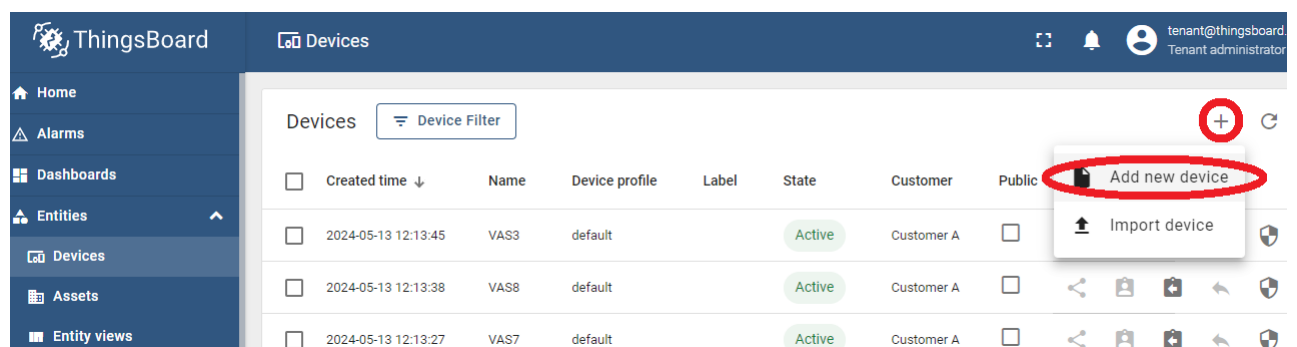


Figure 9. Add new device in the ThingsBoard

Add new device?×

1 Device details

2 Credentials  
Optional

Name\*

VAS2

Label

Device profile\*

default

×

☐ Is gateway

Assign to customer

Description

Next: Credentials

Cancel

Add

Figure 10. Details of the new device

Device created. Let's check connectivity!×

HTTP

MQTT

CoAP

Use the following instructions for sending telemetry on behalf of the device using shell

Windows

MacOS

Linux

Install necessary client tools

Starting Windows 10 b17063, cURL is available by default

Execute the following command

curl -v -X POST http://thingsboard.vas.internal:8080/api/v1/KliY8f

State Inactive

Latest telemetry

Time	Key	Value
<div><div></div><div>No latest telemetry</div></div>		

☐ Do not show again

Close

Figure 11. “Device created” window

## LoRaWAN Stack for VAS sensors

### Get access token of the ThingsBoard device

- Open Web GUI of the ThingsBoard.
- Navigate to: Entities → Devices
- Select previously created device (e.g. VAS2) i.e. click anywhere on the line of this device. After that, a window with details of this device should open, as shown in Figure 12.
- Press “Copy access token” button. Save it somewhere.
- Example of the access token: AbA1NTFBYStefSgFVVXm

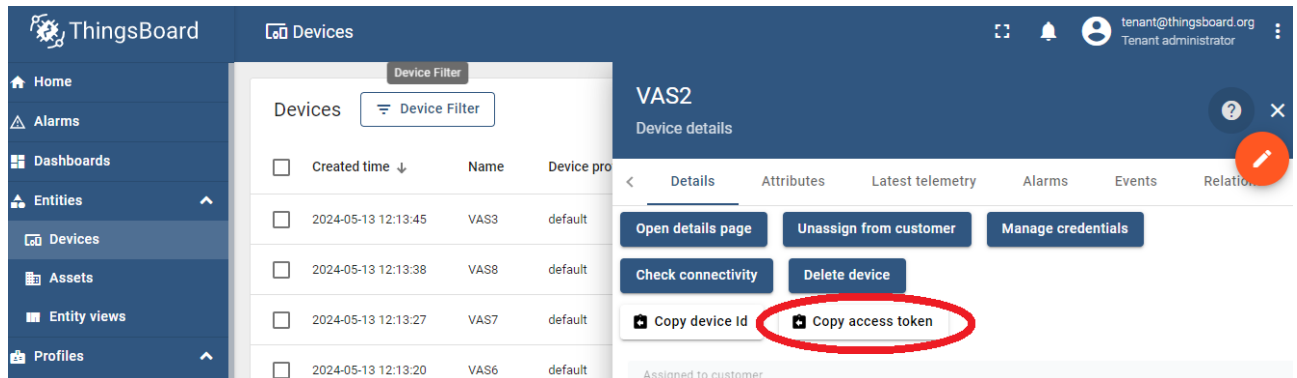


Figure 12. Details of the ThingsBoard device

### Set Device access token in ChirpStack:

- Open Web GUI of the ChirpStack by loading following web page:
  - <https://chirpstack.vas.internal/>
- OR:
  - [http://<IP\\_OF\\_THE\\_SERVER>:8080/](http://<IP_OF_THE_SERVER>:8080/)
- Navigate to the configuration of the VAS device, e.g.
  - Tenant → Applications → VAS Application → VAS device (e.g. VAS2) → Configuration
- Open Variables menu entry, as shown in Figure 13 and press “Add variable” button
- Create following variable:
  - Key: ThingsBoardAccessToken
  - Value: Access token of the ThingsBoard device, e.g. AbA1NTFBYStefSgFVVXm
- Press “Submit” button.

After that, device should connect to the ThinsBoard and its state should be “Active”, as shown in Figure 14.



LoRaWAN Stack for VAS sensors

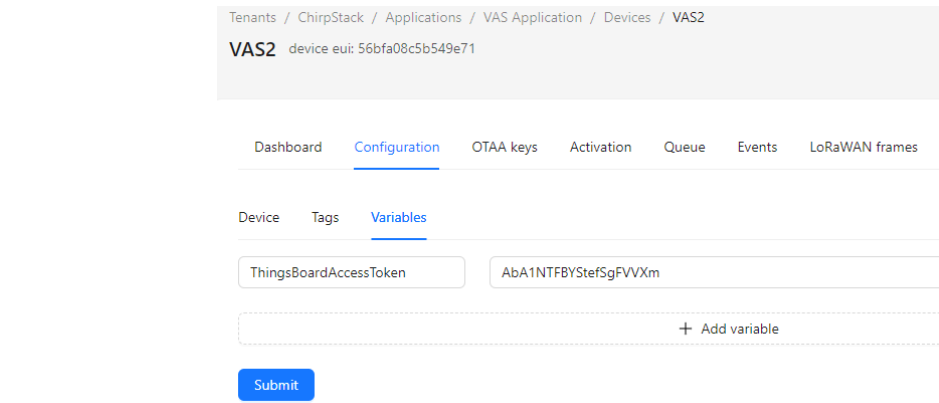


Figure 13. Variables of the ChirpStack device

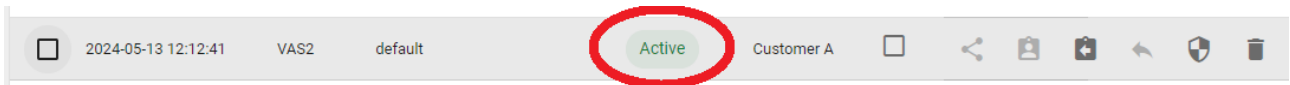


Figure 14. Active ThingsBoard device

# 3

## Configuration of the LoRaWAN Gateway (RAK7289CV2 WisGate Edge Pro)

Manufacturer's quick start guide is available at:

<https://docs.rakwireless.com/Product-Categories/WisGate/RAK7289-V2/Quickstart/>

### 3.1 Initial configuration

Power on Gateway using included PoE Injector.

Wi-Fi AP Mode is used to make initial configuration of the gateway.

Connect to the gateway's default Wi-Fi network.

By default, gateway will create a Wi-Fi Network named RAK7289\_XXXX, where "XXXX" is the last two bytes of the Gateway MAC address.

Open any Web Browser and navigate to the following address: <https://192.168.230.1/>

Web GUI of the Gateway should open.

Set password for access to the gateway's Web GUI as shown in Figure 15.

- Default user is: root

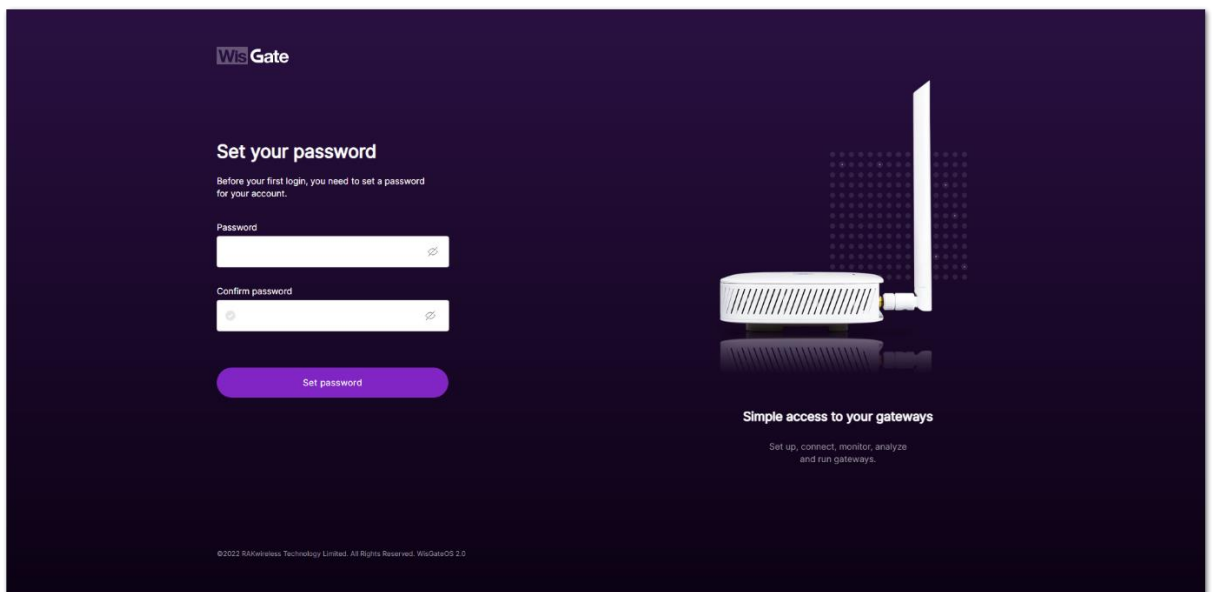


Figure 15. Set password on the RAK Gateway

After setting the password, the dashboard of the gateway should open.

If needed, you can click on the WisGate logo at the top left corner of the Web GUI, to show names of the menu entries, as shown in Figure 16.

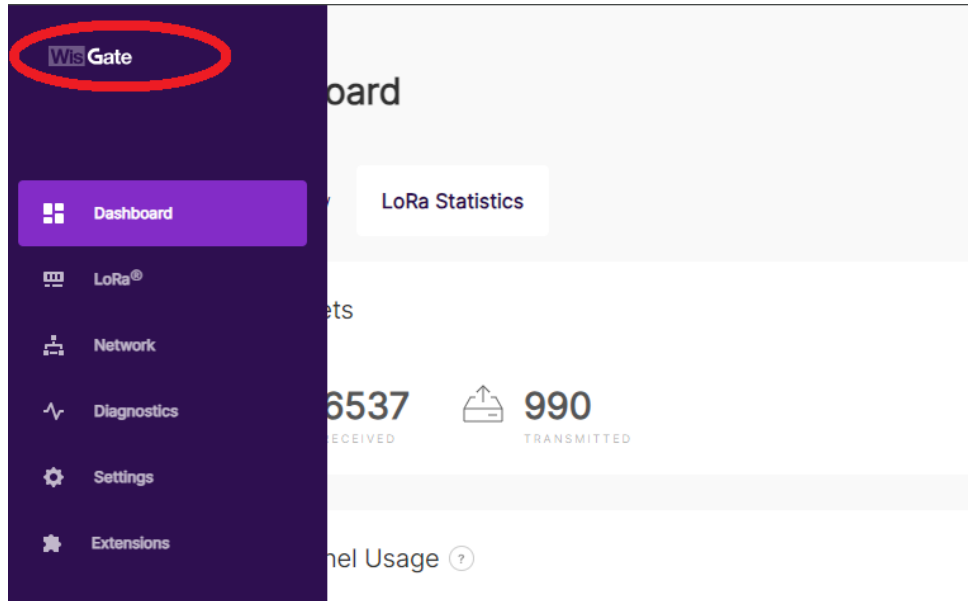


Figure 16. Press on WisGate logo, to show names of the menu entries

Set Wi-Fi password:

- Navigate to: Network → LAN (at the top of the network menu) → Expand (press arrow down) Wi-Fi entry → Settings
- Settings of the Wi-Fi interface should open, as shown in Figure 17. Change following settings:
  - Encryption: WPA2-PSK
  - Key: <YOUR\_PASSWORD>

Figure 17. Settings of the Wi-Fi interface

Set static IP on the LAN interface:

- Navigate to: Network → WAN (at the top of the network menu) → Expand (press arrow down) Ethernet entry → Settings
- Settings of the Ethernet interface should be opened as shown in Figure 18.
  - Enable WAN and disable LAN: enabled
  - Static address: selected
  - IPv4 address: <YOUR\_IP>
  - IPv4 netmask: <YOUR\_NETMASK>
  - IPv4 router: <YOUR\_DEFAULT\_GATEWAY>
  - DNS Server: <YOUR\_DNS\_SERVER>
    - NOTE: Don't forget to press Add button after entering DNS address.

Interface

Enable WAN and disable LAN

Protocol

Static address DHCP client PPPoE

IPv4 address

192.168.10.100

IPv4 netmask

255.255.255.0

IPv4 router

192.168.10.100

Use custom DNS servers

192.168.10.1 x

DNS Server

Add

Figure 18. Settings of the Ethernet interface

Update Firmware

- Navigate to: Settings → Firmware
- Check currently installed version of the firmware. In the example, shown in Figure 19, current version is 2.2.2. Also, the version of the firmware should be shown at the bottom of the Web GUI.

Firmware

Current version

WisGateOS\_2.2.2\_RAK

Upload the new firmware file

Drop your RWI file here or  
[choose file](#)

☒ Keep settings after updating

Update

Figure 19. Current version of the firmware

- Check latest available firmware and release notes at the following web page: <https://downloads.rakwireless.com/#LoRa/WisGateOS2/>
- If newer firmware is available download and unzip it.
- Then navigate to the firmware page on the Web GUI (Settings → Firmware) and upload the new firmware file (.rwi), enable “Keep setting after updating” option and make an update.
- Wait for the update to finish.

Take a note of the Gateway's EUI:

- EUI is needed to connect LoRaWAN gateway to the ChirpStack.
- Navigate to: Dashboard → Overview
- Various parameters of the gateway is shown, as can be seen in the example provided in Figure 20.

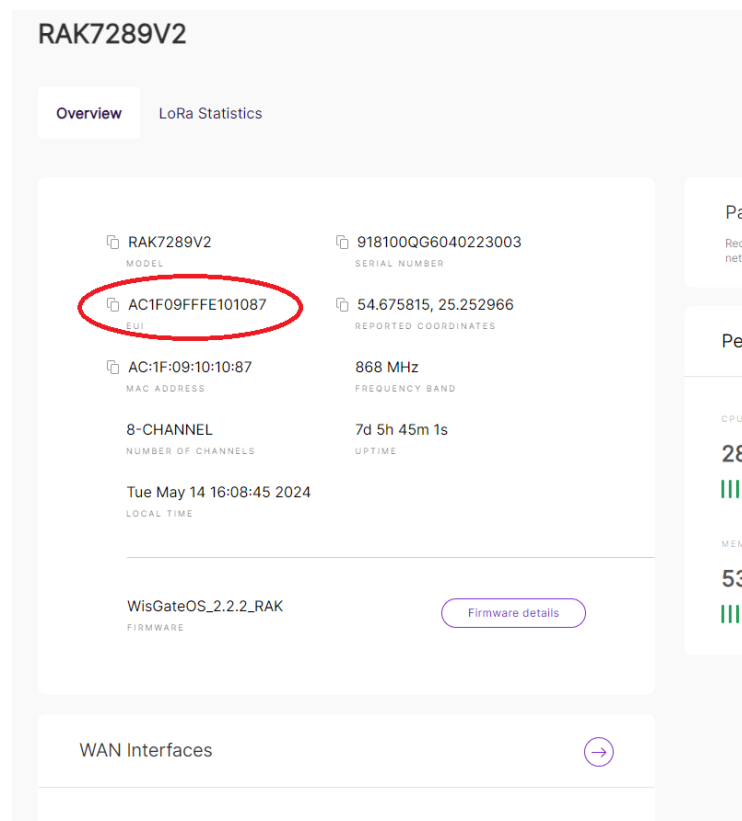


Figure 20. EUI of the Gateway in the Overview page

- As it is seen, in this example EUI is: AC1F09FFFE101087

### 3.2 Configure LoRa settings

Main LoRa settings are shown in Figure 21.

Open Web GUI of the Gateway.

Open LoRa settings

- Work mode: Packet forwarder
- Log Level: NOTICE
- Frequency Plan:
  - Country: Lithuania
  - Regio: EU868
- Protocol:

- Protocol: LoRa Gateway MQTT Bridge
- Statistic interval (s): 30
- LoRa Gateway MQTT Bridge Parameters
  - MQTT Protocol: MQTT for ChirpStack 4.x (protobuf)
  - MQTT Broker Address: chirpstack.vas.internal
  - MQTT Broker Port: 8883
  - MQTT Version: 3.1.1
  - QoS: 1 - At Least Once
  - Keepalive interval (s): 10
  - Clean session: enabled
  - Retain: disabled
  - Enable User Authentication: disabled
  - SSL/TLS Mode: Self-signed server & client certification
  - TLS Version: TLS v1.2
  - CA certificate: upload **ca.pem** generated in Chapter 2.5
  - Client certificate: upload **client\_cert.pem** generated in Chapter 2.5
  - Client key: upload **client\_key.pem** generated in Chapter 2.5

The screenshot displays the configuration interface for the LoRaWAN gateway, divided into two main sections: 'Protocol' and 'LoRa Gateway MQTT Bridge Parameters'.

**Protocol Section:**

- Choose from the available protocols.
- Protocol: ☒ Semtech UDP GWMP Protocol, ☒ LoRa Gateway MQTT Bridge
- Statistic interval (s):

**LoRa Gateway MQTT Bridge Parameters Section:**

- MQTT Protocol:
- MQTT Broker Address:
- MQTT Broker Port:
- MQTT Version:
- QoS:
- Keepalive interval (s):
- ☒ Clean session, ☐ Retain
- ☐ Enable User Authentication
- SSL/TLS Mode:
- TLS Version:
- CA certificate:

Figure 21. Main Settings of the LoRaWAN gateway

### 3.3 Edit hosts file on the gateway

At the moment of writing this document, there is a bug in the gateway's firmware – it does not use custom DNS server, therefore, it can't resolve internal domain name, used for LoRaWAN server, i.e. chirpstack.vas.internal.

As a workaround, this domain name can be fixed by editing hosts file on the gateway.

## LoRaWAN Stack for VAS sensors

By default, SSH is enabled on the gateway. Therefore, use any SSH client to connect to the gateway and login using default username (root) and password, which was set during the initial configuration of the gateway. E.g.:

```
ssh root@<IP_OF_THE_GATEWAY>
```

Open hosts file using vi editor:

```
vi /etc/hosts
```

Press INSERT button on the keyboard.

Append entry for the domain name of the LoRaWAN server (replace the IP with the correct one):

```
#ChirpStack, mosquitto server
```

```
<IP_OF_THE_GATEWAY> chirpstack chirpstack.vas.internal mosquito
```

Save and quit vi editor by pressing typing following combination:

```
:wq
```

## References:

- [1] RAK Wireless, “8 or 16 channel Outdoor LoRaWAN Gateway.” Accessed: Apr. 26, 2024. [Online]. Available: <https://store.rakwireless.com/products/rak7289-8-16-channel-outdoor-lorawan-gateway?variant=42334687789254>
- [2] ChirpStack, “ChirpStack, open-source LoRaWAN® Network Server.” Accessed: Apr. 26, 2024. [Online]. Available: <https://www.chirpstack.io/>
- [3] The ThingsBoard Authors, “ThingsBoard. Open-source IoT Platform.” Accessed: Apr. 26, 2024. [Online]. Available: <https://thingsboard.io/>
- [4] The ThingsBoard Authors, “Installing ThingsBoard using Docker (Linux or Mac OS).” Accessed: Apr. 26, 2024. [Online]. Available: <https://thingsboard.io/docs/user-guide/install/docker/>
- [5] Traefik Labs, “Traefik, The Cloud Native Application Proxy.” Accessed: Apr. 26, 2024. [Online]. Available: <https://traefik.io/traefik/>