User Guide

PoE enabled Raspberry Pi 3B+ based LoRaWAN™ Gateway

LRWGWx-RPI-xxx

Table of Contents

! The device must not be used without a proper 50 Ohm Antenna on the SMA port (RF IO). So before powering the device make sure an antenna is connected.

Overview

The LRWGW RPi, in the following referred to as Gateway, is a LoRaWAN Gateway based on a Raspberry Pi 3B+, a LRWCCx-MPCIE Concentrator Card and a SBCPoE RPi adapter hat. The latter is hosting the concentrator card and provides connectivity to the RPi UART port and it acts as power supply for the whole system when attached to a PoE Power Sourcing Equipment (PSE).

Hardware Installation

The Gateway has two latches with two holes each that can be used to mount the Gateway on a wall. Try to install the gateway in an open place high above ground if possible and try to avoid rooms with concrete walls. The antenna should be aligned upright.

! The device is only suited for indoor use.

Powering Up

The Gateway can be powered either by the micro USB port or through the Ethernet port when connected to a Power Sourcing Equipment like a injector, midspan or switch.

! Make sure to secure the Raspberry Pi OS by changing the default password as described in the Securing your Raspberry Pi article.

Network Setup

If not yet done, connect an Ethernet cable to connect the Gateway to your local network. The gateway does the following to connect itself in the network with IPv4:

- It tries to obtain an IP address, default gateway and DNS servers via DHCP.
- It reports the host name `raspberrypi` back to the DHCP server which is used to register it in the local DNS server if supported.

Optionally it is possible to connect the Gateway to the local network via WiFi but this is out of scope for this user guide.

If the Gateway is integrated in the network properly, a remote log-in via SSH is possible with the following Raspberry Pi OS default credentials:

User:	`pi`
Password:	`raspberry`

When logged in remotely via SSH, the DNS resolution and Internet connectivity should be tested via:

```
ping www.google.com
```

The output should look similar to this:

```
PING www.google.com(fra15s24-in-x04.1e100.net (2a00:1450:4001:820::2004)) 56 data bytes
64 bytes from fra15s24-in-x04.1e100.net (2a00:1450:4001:820::2004): icmp_seq=1 ttl=57 time=8.60 ms
64 bytes from fra15s24-in-x04.1e100.net (2a00:1450:4001:820::2004): icmp_seq=2 ttl=57 time=7.19 ms
64 bytes from fra15s24-in-x04.1e100.net (2a00:1450:4001:820::2004): icmp_seq=3 ttl=57 time=7.93 ms
^C
--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 7.191/7.908/8.603/0.581 ms
```

OS Image

The OS is a slightly modified Raspberry Pi OS Buster Lite as of August 2020 stored on a 16 GiB SD Card. The OS has been modified to prevent write access to the SD card as much as possible to assure its longevity. Furthermore the required software to act as a LoRa Gateway has been installed. Below a complete list of all applied changes:

- The HAL and packet forwarder from Semtech has been installed as deb packages named `picogw-hal` and `picogw-packet-forwarder`
- A Systemd unit `lora_pkt_fwd` for supervising and launching the packet forwarder process automatically is enabled
- The SSH server is enabled
- Enabled read-only operation using `overlayfs`. By specifying `init=/sbin/reinit` in `/boot/cmdline.txt`.
- Enabled UART serial for LoRa use. Specified `enable_uart=1` in `/boot/config.txt`. Removed `console=serial0,115200` from `/boot/cmdline.txt`.
- Enabled i2c for temperature sensing. Specified `dtparam=i2c_arm=on` in *`/boot/config.txt`*. Installed `i2c-tools`. Loads `i2c_dev` module on boot.

List of custom installed files can be obtained with:

```
dpkg -L picogw-packet-forwarder picogw-hal reinit-overlayfs nfuse-tweaks
```

Toggle Read-Only Operation

In RO mode, any changes within root partition are allowed, without ever writing to SD card, that is so, because changes are kept in RAM, and will be lost after a reboot. Thus before applying updates or modifying configuration files, device must be rebooted in RW mode.

Make sure to re-enable read-only mode after you're done with modifications.

To enable RW mode, where changes are permanent: `sudo enableWriteAccess.sh`
To enable RO mode, where changes are temporary: `sudo disableWriteAccess.sh`
Please note, switching between modes implies a reboot.

After switching, you can determine which mode you are in depending on whether there is output produced by the command below, no output means RW mode, and vice versa:

```
mount | grep overlay
```

Example output of RO mode would be:

```
/dev/mmcblk0p2 on /overlay/lower type ext4 (ro,...)
none on /overlay/lower/overlay type tmpfs (rw,...)
overlay on / type overlay (rw,...,lowerdir=/,upperdir=/overlay/upper/data,workdir=/overlay/upper/work)
```

Whereas no output would imply RW mode. Likewise mount entry of root is:

```
/dev/mmcblk0p2 on / type ext4 (rw,noatime)
```

Disk Partition Expansion

The Debian based Raspberry Pi OS provides a method of expanding the root filesystem via raspi-config, described in the official raspi-config guide.

Make sure you're in RW mode, and enter the Raspberry Pi OS configurator:

```
sudo raspi-config
```

Then navigate the menu appropriately (note there won't be a confirmation dialogue):

1. Advanced Options
2. Expand Filesystem

Then reboot.

Software Upgrade

The Debian based Raspberry Pi OS can be easily upgraded as described in the official update and upgrade guide.

Before updating do enable RW mode, otherwise changes to root partition will be lost (command reboots device):

```
sudo enableWriteAccess.sh
```

Update your system:

```
sudo apt-get update
sudo apt-get -y dist-upgrade
```

Reboot back into RO mode:

```
sudo disableWriteAccess.sh
```

! Make sure to establish a regular procedure to update the base OS to minimize the risk of a security hole being exploited.

LoRa Packet Forwarder Setup

Changing the concentrator’s radio related configuration might void the operating permit in the regulatory domain your are operating the Gateway.

Depending on whether device contains card with European frequency 868Mhz or US frequency 915Mhz, templates are provided at paths below:

lrwgw8-rpi-868:	`/usr/share/doc/lora-pkt-fwd/conf/global_conf_PicoV1p0_EU.json`
lrwgw8-rpi-915:	`/usr/share/doc/lora-pkt-fwd/conf/global_conf_PicoV1p0_US.json`

Copy appropriate template file into `/etc/lora_pkt_fwd/global_conf.json`:

```
sudo cp /usr/share/doc/lora-pkt-fwd/conf/global_conf_PicoV1p0_XX.json \
    /etc/lora_pkt_fwd/global_conf.json
```

The LoRaWAN Concentrator Card has a unique ID given at production. This ID can be used as a 64-bit MAC address. Which must be provided in the `gateway_conf.gateway_ID` property within `/etc/lora_pkt_fwd/global_conf.json` file. ID can be obtained from the device via:

```
sudo util_chip_id -d /dev/serial0
```

Which may return the unique ID:

```
1234567890ABCDEF
```

The configuration file would look similar to this:

```
...
"gateway_conf": {
    "gateway_ID": "1234567890ABCDEF",
    /* change with default server address/ports, or overwrite in local_conf.json */
    "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
}
```

A file called `/etc/lora_pkt_fwd/local_conf.json` can be made, whose properties would override ones at `/etc/lora_pkt_fwd/global_conf.json`.

Once configuration is done, enable the service to auto-start on boot up:

```
sudo systemctl enable lora_pkt_fwd.service
sudo systemctl start lora_pkt_fwd.service
```

Reading the Temperature Sensor

The devices comes with an LM75B temperature sensor attached to the I2C bus of the Raspberry Pi. It can be read out in degree Celcius with the following command:

```
i2cget -y 1 0x48 0x00 w | awk '{printf("%.1f C.\n",
(a=((("0x"substr($1,5,2)substr($1,3,1))*0.0625)+0.1) )>128?a-256:a)}'
```

Securing the Gateway

Make sure to secure the Raspberry Pi OS by changing the default password as described in the Securing your Raspberry Pi article.

n-fuse RESERVES THE RIGHT TO CHANGE PRODUCTS, INFORMATION AND SPECIFICATIONS WITHOUT NOTICE. Products and specifications discussed herein are for reference purposes only. All information discussed herein is provided on an "AS IS" basis, without warranties of any kind. This document and all information discussed herein remain the sole and exclusive property of n-fuse. No license of any patent, copyright, mask work, trademark or any other intellectual property right is granted by one party to the other party under this document, by implication, estoppel or other-wise. n-fuse products are not intended for use in life support, critical care, medical, safety equipment, or similar applications where product failure could result in loss of life or personal or physical harm, or any military or defense application, or any governmental procurement to which special terms or provisions may apply. For updates or additional information about n-fuse products, contact us. Product names, logos, brands, and other trademarks featured or referred are the property of their respective trademark holders.

Version 14.10.2020