

Meshtastic \rightarrow AREDN

Jed Marti KI7NNP

January 2, 2025

Abstract

I describe software to transfer Meshtastic text messages to the AREDN digital network and the internet. This is primarily a demonstration that this is possible, not a finished product.

1 Introduction

I describe 3 different programs useful for some emergency communication situations. Four components are important.

Meshtastic transceiver 915 MHz band for transferring short messages in a mesh network. These operate with a modulation scheme known as LoRa.

Portable Text Control device Connected to the Meshtastic transceiver to send enter and display text messages. Typically this is a cell phone or tablet.

Fixed Meshtastic/AREDN base station This has a Meshtastic transceiver and an AREDN SHF digital transceiver. A laptop or Raspberry PI running Linux are required.

Optional: Internet access Meshtastic messages can be directed to an AREDN node or the internet.

There are many limitations:

1. Messages from AREDN or the internet cannot be transferred to Meshtastic devices (software bug).
2. Multi-hop messages: Meshtastic \rightarrow AREDN \rightarrow Internet are not implemented.
3. The AREDN mail server has many limitations.
4. AREDN mail service is possible, that is AREDN \rightarrow AREDN is weak.
5. Due to limitations in the Windows operating system, most of the code runs only under Linux.
6. The AREDN target IPV4 number must be known by the base station.
7. The Internet SMTP mail server IPV4 number and account information must be known by the base station.
8. There is no connection to lower frequency message passing schemes such as Winlink.
9. The system and its components are not “plug-and-play” and require training and regular exercise to be useful.

2 The Scenario

In an emergency situation a number of low level operators (a CERT team for example) would report damage and casualties via text messages to a central site where they are passed automatically to a higher level. The range of the base LoRa modulation scheme is potentially much greater than other systems such as GMRS. Likewise, message content does not need to be transcribed (and garbled) at the first communication hop. Important information such as GPS location will automatically be included in messages without user intervention. Unless a high-gain antenna is used, the mobile operators do not need an amateur license.

It remains to be proven that LoRa does in fact have a longer range than GMRS.

3 Meshtastic to Digital

Meshtastic devices communicate with the rest of the world through an Internet of Things (IOT) protocol named MQTT (Message Queuing Telemetry Transport) together with messages described by protoBuf. The communication is through Bluetooth, WiFi, Ethernet, or serial port. Various applications on Android devices, Apple, Windows, and Linux provide varying levels of support.

I therefore developed a program unimaginatively called **mqtt** to read packets from ethernet or WiFi and communicate with other network systems, in particular, AREDN meshes and the internet. You run **mqtt** continuously and it automatically reads messages from a Meshtastic device and decides what to do with them. You can also run the webserver or Bluetooth interfaces to control and monitor the Meshtastic device.

3.1 Setup Meshtastic Base Station

This is for the base station only. Do not pass this on to stations without an MQTT server.

3.1.1 Setting up Two Channels

Get to the **Channels** option. You should see 8 listed, we're modifying the first 2.

1. **Set up Channel #0** Using your favorite interface set the first channel as follows:

pre-Shared Key Delete all the characters. Do not click **Generate**.

Name For now this should be **LongSlow**, no spaces.

Uplink Enabled *Send messages from the local mesh to MQTT*. If the button is to the right, messages to this channel will be sent to the PI for processing.

Downlink Enabled *Send messages from MQTT to the local mesh.* Doesn't matter, can't make it work.

Submit Click **Submit** or what ever your interface wants.

2. **Set up Channel #1** Switch to channel #1. Set the following.

Role Select **Secondary**

pre-Shared Key Delete all the characters. Do not click **Generate**.

Name Set this to **ARESMAIL**. Messages sent to this channel will be sent to the MQTT server for mail processing.

Uplink Enabled The toggle must be to the right for messages to be sent to the MQTT server.

Downlink Enabled Doesn't matter doesn't seem to work.

The rest ... Doesn't matter.

Submit Click **Submit** or what ever your interface wants.

3.1.2 Radio Configuration

Next go to the radio configuration section and look for **Network**. Enable WiFi with your network name or enable the Ethernet port if that's your thing. Use DHCP as you may get your assigned address from an AREDN router.

If you have an internet connection, the NTP server will be used to get the time from your network connection.

In the **LoRa** settings make sure of the following:

Region If you're in the US, make sure it's set for US.

Hop Limit This should be 3.

Frequency Slot This should be 59, where amateur higher power is allowed.

Ignore MQTT *Don't forward MQTT messages over the the mesh.* This should probably be off but the Meshtastic folks have a bizarre problem with double negatives.

OK to MQTT Probably should be true so other stations running MQTT can fiddle with their copies.

Use Preset *Use one of the predefined modem presets* Yes, push slider to the right.

Modem Preset *Modem preset to use.* For now select **Long Slow**. This will modify **Bandwidth**, **Spreading Factor** and **Coding Rate** which will read 0 so don't touch these.

Transmit Enabled *Enable/Disable transmit (TX) from the LoRa radio.* Slide this to the right and ignore the assault on English.

Frequency Offset Leave this alone unless you know what you're doing, I don't.

Boosted RX Gain Slide button to the right, it might help.

Override Frequency This needs to be set to 916.625 for unknown reasons.

Done Click the 3.5" floppy disk icon to affect the changes or whatever your interface requires.



Figure 1: The last 3.5” disk I could find

3.1.3 Module Configuration

Now switch to the **Module Config** section and get to the MQTT settings. Do the following.

Enabled *Enable or disable MQTT.* Slide button to the right to enable.

MQTT Server Address Enter the IPV4 number of the PI or laptop you’re using to run the mail server.

MQTT User Name Ignored will be sent during the connection phase.

MQTT Password Ignored but will be sent during the connection phase.

Encryption Enabled Make sure this is off!

JSON Enabled Should be off.

TLS Enabled Make sure this of off.

Proxy to Client Enabled Should be off.

Map Reporting Enabled Can be on or off.

Map Report Publish Interval(s) Set to what you want, but frequent sending will clog the frequency slot.

Approximate Location What ever you want.

Done Click the 3.5" disk icon or what have you (see Figure 1).

It's time to reboot the Meshtastic node. Verify the settings survive the reboot. If you've been using the CLI, this happened on every item you changed.

4 Setting up the MQTT Server - Linux

Download the appropriate MQTT server from <http://www.cog9llc.com/aredn>. The files are named *mqtt_64.tar* and *mqtt_pi.tar*.

```
tar xvf mqtt_64.tar
cd mqtt
sudo make install
```

This moves the executable to */usr/local/bin* or you can put it wherever you want. The *mqtt* directory also includes the source code. To fiddle and recompile:

```
make
```

The executable is **mqtt** and it has a number of command line options.

Short	Long	Value	Description
-a	-addresses	filename	Where address abbreviations are found, - means none (abbrevs)
-ap	-arednPortnum	<i>integer</i>	AREDN port number (3976).
-c	-channel	<i>string</i>	Channel name for mail (ARESMAIL)
-d	-delete		Delete e-mails after sending
-k	-keep	<i>integer</i>	Extra time at startup for watchdog (30)
-L	-log	<i>dirname</i>	Where to put the log file (.log/)
-m	-mail		Start the stdin server
-n	-noise	<i>integer</i>	Set the output noise level (0) 0-5
-p	-portnum	<i>integer</i>	Set listen port number (1883)
-s	-smsdir	<i>dirname</i>	Directory name to put SMS temporaries (/tmp/)
-td		<i>dirname</i>	Directory for packet trace (/tmp/)
-tr			Turn on packet tracing

Table 1: Command line options

A typical startup might be:

```
mqtt -a jedsaddrbook
```

4.1 The Address Book

The address book provides a way of reducing the number of characters required for the to/from fields of a message. Each line begins with the abbreviation followed by either an internet mail address or an IPV4 address of an AREDN mail server. For example:

```
jed      martij@xmission.com
jg       10.59.19.32
a        KI7NNP-A
b        KI7NNP-B
c        KI7NNP-C
ki7mti  ki7mti@gmail.com
```

4.2 Setting up Network E-mail

The system uses *ssmtp* to send mail to a server. In Linux, you need to modify */etc/ssmtp/ssmtp.conf* as a super user. For example, I use X-mission and the file looks something like:


```
#
# Config file for sSMTP sendmail
#
# The person who gets all mail for userids < 1000
# Make this empty to disable rewriting.
root=postmaster

# The place where the mail goes. The actual machine name is required no
# MX records are consulted. Commonly mailhosts are named mail.domain.com
mailhub=mail.xmission.com

# Where will the mail seem to come from?
#rewriteDomain=

# The full hostname
hostname=here give the name of your server

# Are users allowed to set their own From: address?
# YES - Allow the user to specify their own From: address
# NO - Use the system generated From: address
#FromLineOverride=YES
AuthUser=here give your e-mail address, e.g.: martij@xmission.com
AuthPass=here give your password for the internet mail server
#TLS_CA_File=/etc/pki/tls/certs/ca-bundle.crt
```

4.3 Clearing up Router Gateways

If you are servicing both an AREDN net and some other net, the system will be confused about where to send an e-mail. AREDN does not yet have an DNS server that I know of. During startup will examine the current gateway table and warn you if there are two defaults.

The Linux **ip** command shows you the current table. Here there is only one gateway but if you attach another router to the system such as that posed by an AREDN node you will run into trouble.

```
ip r
default via 192.168.1.1 dev enp0s31f6 proto dhcp metric 100
169.254.0.0/16 dev enp0s31f6 scope link metric 1000
192.168.1.0/24 dev enp0s31f6 proto kernel scope link src 192.168.1.226 metric 100
```

If two defaults show up, you must perform the following:

```
sudo ip route delete default
```

4.4 Sending mail from Meshtastic

1. Select the **ARESMAIL** channel.
2. Enter:

```
>to-address<from-address message
```

There should be no spaces except after the *from-address* before the *message* starts. The *to-address* can be a full e-mail address or an abbreviation from the address book. If the address is not an IPV4 number, the program uses **ssmtp** to send it to your mail server.

3. Click or tap send.

5 Setting up the AREDN Mail Server

Download the software from www.cog9llc.com/meshmail_64.tar or www.cog9llc.com/meshmail_pi.tar as appropriate. Then:

```
tar xvf meshmail_xx.tar
cd meshmail
sudo make install
```

This places the **meshmail** program in */usr/local/bin* though you can put it where you like.

The default directory for storing mail is *.meshmail* which you must create. If you're running in your home directory:

```
mkdir .meshmail
meshmail
```

Then in your browser look to *file:///home/your-name/.meshmail*. You will need to periodically refresh your screen, as yet there is no indication of a new e-mail message.