Meshtastic APRS Gateway

What is APRS?

APRS - **Automatic Packet Reporting System** - is an amateur radio protocol designed for real-time tactical digital communications. Originally developed by Bob Bruninga (WB4APR), it carries position data, weather reports, text messages, and telemetry over RF. Reports are aggregated on publicly accessible maps such as aprs.fi, making it a cornerstone of ham radio situational awareness worldwide.

Why Integrate Meshtastic with APRS?

Meshtastic nodes equipped with GPS can be bridged into APRS-IS (the internet backbone of APRS), causing them to appear on aprs.fi maps. This provides several advantages:

- Mesh position data becomes visible to the broader amateur radio community without requiring a dedicated APRS radio.
- Enables interoperability with Winlink gateway operators who already monitor APRS.
- During emergency activations, mesh nodes show up alongside APRS-equipped vehicles and portable stations on the same common operating picture.

License Requirement

Transmitting on APRS RF (144.390 MHz) legally requires an amateur radio license - **Technician class or higher** in the United States, under FCC Part 97. An internet-only APRS-IS feed (the software bridge described here) does not transmit on RF, so it is not license-gated by FCC law. However, APRS-IS network policy still requires you to connect with a valid amateur callsign and a callsign-derived passcode; the passcode system exists to keep APRS-IS restricted to amateur radio use. Receive-only IS access does not require a license, but to inject your nodes' positions you connect with your callsign. Check your national regulations.

aprstastic

The open-source **aprstastic** project (GitHub: `afourney/aprstastic`) is a bidirectional Meshtastic-to-APRS gateway that bridges Meshtastic output to APRS-IS. It runs on any Linux system with internet access - a Raspberry Pi is the most common deployment platform. Verify the current repository and its setup instructions before deploying.

Core requirements:

- Python 3.8+
- An MQTT broker already receiving packets from your Meshtastic node(s)
- An APRS-IS account (callsign + APRS-IS passcode - generated from your callsign)

Setup Overview

1. Install the gateway and its Python dependencies (follow the project's own README; for a manual stack: `pip install meshtastic paho-mqtt aprslib`).
2. Clone the gateway repository and copy the example config file.
3. Edit the config: enter your callsign, APRS-IS passcode, APRS-IS server (e.g. `rotate.aprs2.net:14580`), and MQTT broker address.
4. Test manually using the command documented by the gateway project.
5. Install as a systemd service for automatic start on boot:
   `/etc/systemd/system/aprstastic.service`
   Set `Restart=on-failure` and `RestartSec=10` to recover from transient network drops.

Packet Flow

```
Meshtastic node
 ↓ (LoRa RF)
 Mesh network
 ↓ (LoRa RF)
 MQTT gateway node (USB-connected Pi)
 ↓ (TCP, localhost)
 Mosquitto MQTT broker
 ↓ (TCP, internet)
 aprstastic gateway
 ↓ (TCP, APRS-IS protocol)
 APRS-IS network
 ↓
 aprs.fi (and other APRS clients)
```

Position Filtering and Consent

The gateway config allows you to whitelist specific node IDs or callsigns for forwarding. **Only forward nodes whose operators have consented.** Do not expose the positions of private nodes, shelters, or sensitive installations to the public APRS map without explicit permission from the node owner.

Best practice: maintain a config-controlled allowlist and review it whenever new nodes join your mesh. Note that allowlisting at the APRS gateway only limits APRS exposure - if the source mesh already uplinks to the public MQTT broker, those positions are already public regardless of the APRS allowlist.

Bidirectional Messaging

Some implementations support two-way message bridging - APRS messages addressed to your gateway callsign arrive as Meshtastic messages, and mesh messages can be relayed outbound as APRS messages. This is useful during ARES/RACES exercises where some operators have APRS radios and others have Meshtastic nodes.

Part 97 compliance warning: if you relay Meshtastic content *outbound onto APRS/amateur RF*, FCC Part 97 applies. Encrypted or meaning-obscured traffic is prohibited (§97.113(a)(4)) - so default-encrypted Meshtastic channels cannot lawfully be bridged onto amateur frequencies - and the station must transmit its callsign identification at least every 10 minutes (§97.119). An anonymous or encrypted mesh payload cannot lawfully be retransmitted as amateur traffic.

Implementation complexity is higher than position-only bridging; test thoroughly before deploying in an operational scenario.

Frequency Note

APRS in North America operates on **144.390 MHz** (a 2-meter FM frequency). This is entirely separate from LoRa mesh frequencies (915 MHz in North America). The APRS gateway described here is a *software bridge over IP* - it does not require your Pi to transmit on 144.390 MHz. The RF-to-internet connection is handled by existing APRS-IS infrastructure.

If you want true RF APRS transmission (e.g. for locations without internet), you need a licensed APRS radio and a TNC - that is a separate project beyond this software gateway.

Revision #3
Created 2026-05-03 04:10:57 UTC by Mesh America Admin
Updated 2026-06-09 01:58:16 UTC by Mesh America Admin