Growing Your Network

How to recruit hosts, run community events, document your network, and manage growth at scale.

- Running Mesh Events and Demonstrations
- Network Documentation Standards
- Handling Network Growth and Congestion

Running Mesh Events and Demonstrations

The fastest way to grow a mesh community is to let people experience the network firsthand. Events and demonstrations convert curious bystanders into active participants in a way that no amount of documentation can match.

The Mesh Meetup Format

A mesh meetup requires no special venue and minimal equipment. The basic format:

1. Gather 5 - 10 people at a park or open community space.
2. Each participant brings or borrows a Meshtastic node.
3. Power them on, watch the mesh form in real time.
4. Run through the basics: send a message, observe who receives it, check the map to see each other's positions, walk to the edge of the park to test range.

The goal is not to teach technical details - it's to create a "this actually works" moment that participants will describe to others. Keep it relaxed. Keep it short. Follow up with a Discord or group message so attendees can stay connected.

Integration with Existing Events

You do not need to organize a standalone event to demonstrate the mesh. Bring it to events that already draw the right audience:

- **Ham radio field days** - A natural fit. Many operators have never seen Meshtastic; a working demonstration at a field day generates more interest per hour than almost any other venue.
- **CERT and ARES exercises** - Emergency preparedness audiences respond strongly to a practical off-grid messaging tool. You do not need a full exercise; a side-table demo during setup or break time is enough.
- **Maker faires** - Technically curious attendees who enjoy seeing novel hardware in action.
- **Outdoor clubs (hiking, camping)** - Groups that go into areas with no cell coverage are interested in off-grid messaging. Be clear in demos that mesh is best-effort and, without nearby repeaters, only short-range between handheld nodes (often well under a few miles

in terrain) with no guaranteed delivery and no connection to emergency services. It is not a substitute for a satellite messenger or personal locator beacon (PLB) in a life-safety situation.

Bring 3 - 5 pre-configured nodes. Let people pick them up and try them. The hands-on experience is the entire pitch.

EmComm Exercise Integration

One of the highest-value moves for community credibility is proposing a mesh communications component to your county's next emergency management tabletop exercise or ARES drill. Contact your county emergency manager or ARES/RACES leadership and offer to run a Meshtastic messaging segment.

Demonstrating value in a formal exercise context creates buy-in from officials who influence future infrastructure decisions. An emergency manager who has seen the mesh work in a controlled setting is far more likely to support permanent installations on municipal buildings.

Demo Kit

Maintain a dedicated loaner kit for events:

- 3 - 5 nodes, all pre-configured with your local channel settings
- Charged batteries or power banks for each node
- A one-page quick-start card: how to turn it on, how to send a message, what the map view shows
- A way to track who has each device (a simple sign-out sheet is enough)

Get the kit back at the end of every event. Pre-configured nodes that leave your possession and are never returned are nodes that may end up causing interference or confusion on the network later.

Online Presence

A simple online presence makes it dramatically easier for newcomers to find your local mesh community, learn the channel settings, and get involved:

- **Discord server** - Many cities have active Meshtastic Discord channels. Search before creating a new one. If one exists, become active there; if not, a new server is easy to set up.

- **Simple website** - Even a single static page with your channel name, modem preset, and a contact link is sufficient. New participants should be able to find this without joining a private group first.

Don't publish the channel PSK publicly; share it via QR code to participants you trust. But understand what the PSK does and does not do: it provides message *confidentiality* (outsiders without the key can't decode traffic), not access control or sender authentication. Because every participant holds the same shared key, anyone who has it can read and even forge messages on the channel, so distributing it to many "verified" participants offers only limited security. Don't treat a shared PSK as making the channel truly secure.

Measuring Success

Track concrete, observable metrics rather than relying on impressions:

- **Node count on meshmap.net** - Check month over month. A growing trend is the clearest indicator that the community is healthy. (Note: meshmap.net only shows nodes that report to the public Meshtastic MQTT server, so it undercounts a private community network - use your own node database or app for an accurate count.)
- **Active repeater count** - Nodes that are consistently online over multiple weeks, not just at events.
- **Set written goals** - "10 active nodes in the metro area by year end, 3 hilltop repeaters." Goals create shared direction and make progress visible.

Celebrate milestones publicly in your Discord or website. Recognition keeps volunteers engaged.

Network Documentation Standards

A community mesh network that isn't documented is one resignation or one hardware failure away from being unrecoverable. Documentation is not overhead - it's the infrastructure that makes the physical infrastructure maintainable.

What to Document

For every permanent repeater, record:

- **Location** - Address or GPS coordinates, plus a plain-language description (e.g., "roof of First Baptist Church, southwest corner")
- **Hardware** - Device model, enclosure type, antenna type and gain
- **Firmware version** - Current version and date of last update
- **Channel configuration** - Channel name, modem preset, and whether a custom PSK is in use (do not record the PSK in shared documentation)
- **Power system** - Solar + battery, wall power, PoE. Battery capacity if applicable.
- **Maintainer contact** - Name and contact method for the person responsible for this node

Beyond individual nodes, document:

- **Network topology** - Which repeaters link to which under typical conditions. A simple diagram is worth pages of prose.
- **Channel settings** - The canonical record of your community's channel name and modem preset. This is what you share with newcomers.
- **Coverage map** - Based on tested range, not theoretical propagation. Mark areas where coverage has been verified by walking tests or field days.

Using the Library

Keeping node information in a shared wiki rather than a single operator's notes means the network survives personnel changes. Maintain a repeater reference entry for every permanent node in your network, capturing the fields listed above.

Update the library entry whenever hardware changes, firmware is updated, or maintainer contact information changes. Stale documentation is worse than no documentation because it misleads the next person who needs to act on it.

Repeater Naming Conventions

Choose a consistent naming scheme before you deploy your first permanent repeater. Changing it later is painful - nodes appear on maps and in message logs, and inconsistency creates confusion. Common options:

- **Location-based**: `MeshAmerica-WestHills-1` - Human-readable, easy to orient on a map
- **Callsign-based**: `KD9XYZ-SiteA` - Ties a node to a licensed amateur. Use this ONLY if the node operates under amateur (Part 97) rules, which means open, unencrypted operation with no obscured content. Do NOT put a callsign on a node running a private/PSK-encrypted channel, or on a default Part 15 node on 915 MHz ISM - mislabeling such a node with a callsign can imply unlawful amateur operation. Callsign-as-identity only provides "EmComm accountability" in unencrypted Part 97 mode.
- **Sequential**: `PDX-Repeater-01` - Simple, scales cleanly

Pick one scheme and document it. New nodes added by new community members should follow the same convention.

Channel Documentation

Publish your channel name and modem preset publicly - on your Discord, website, or wiki. New participants need this information to join the network.

Do **not** publish the PSK publicly if your community uses a private channel for security. Share the PSK via QR code (or channel URL) to verified participants only - the QR/URL carries the PSK, which is why it must be shared carefully. The QR code / channel URL encodes the shared channel(s) you choose to include plus the LoRa/modem-preset settings - it is the standard mechanism for onboarding new nodes. Review which channels are included before sharing it.

Firmware Version Tracking

Maintain a log of when each node was last updated. Meshtastic firmware is actively developed; nodes running outdated versions may miss bug fixes, compatibility improvements, or security patches.

Schedule a quarterly firmware review: check each node's current version, identify which nodes need updates, and coordinate with site maintainers to get access for updates. A node that hasn't been updated in over a year should be treated as a maintenance priority.

Incident Log

When a node goes offline - whether due to weather, power failure, hardware failure, or vandalism - document:

- What happened
- When it was detected
- What was done to restore service
- What (if anything) changed to prevent recurrence

This institutional knowledge prevents repeat failures. The second time a solar controller fails in a cold snap, the person responding should be able to find a record of what worked last time.

Succession Planning

Community networks outlast individual operators. Plan for turnover:

- Ensure at least two people have physical and administrative access to every critical node.
- Document access procedures (who has the key to the rooftop, who has the admin channel QR code).
- Store admin channel QR codes securely, with a trusted backup outside your own household. A fireproof envelope with a co-maintainer is a reasonable standard.
- Review the succession plan annually. People move, change jobs, and lose interest. The plan is only useful if it's current.

Handling Network Growth and Congestion

A Meshtastic network that works well with five nodes may behave poorly at fifty. Managing growth proactively - rather than reacting after congestion degrades performance - is what separates a durable community network from one that works great until it doesn't.

Early Stage: 1 - 10 Nodes

At this scale, almost any configuration works. Long Fast is a reasonable default - it has good range and is the most common preset, making it easy for newcomers to join.

Focus at this stage belongs on placement, not optimization. A single well-placed repeater on a hilltop or water tower does more for network coverage than ten ground-level nodes. Get repeaters in good locations first.

Growth Stage: 10 - 50 Nodes

As the network grows, start monitoring channel utilization. The Meshtastic app displays this metric; check it periodically. Channel utilization is generally healthy below ~25% - the firmware defers transmitting until utilization drops below 25% (packets are deferred, not dropped). As a rule of thumb on the app's 1-minute window, green is under 25%, orange is 25-50%, and red is over 50%. Sustained utilization above ~25% (and especially above 50%) is a warning sign.

Actions at this stage:

- Audit hop limits. High hop limits cause individual messages to generate multiple retransmissions across many nodes. A hop limit of 3 is appropriate for most community networks; higher values compound congestion (the Meshtastic maximum is 7).
- Review node density. Clusters of nodes in the same small area all hear and retransmit each other's traffic. Nodes that add density without extending coverage add congestion without benefit.
- Begin discussing a potential preset migration with the community before it becomes urgent.

Scaling Stage: 50+ Nodes

At this scale, Long Fast will likely show sustained high utilization. Seriously consider migrating to Medium Fast or Medium Slow, which use higher data rates and shorter airtime per message to reduce congestion - in exchange for some range and receiver sensitivity compared to Long Fast. The point of the migration is to gain channel capacity, not range: Long Fast actually reaches farther, while the Medium presets are faster but shorter range. That capacity tradeoff is what matters when node count is high and airtime is the remaining problem.

This migration must be coordinated as a community. **All nodes must change simultaneously, or you split the network** - nodes on different presets cannot communicate.

Preset Migration Process

1. **Pick a date and time** for the migration window.
2. **Announce in all community channels** at least two weeks in advance. Post in Discord, on the website, and in any mailing lists. Include the specific preset being adopted.
3. **Document the change procedure** - a step-by-step guide for changing the preset in the Meshtastic app, for both Android/iOS and the web client.
4. **Change infrastructure repeaters first.** These are the reference points that participants will see when they join on the new preset.
5. **Ask all participants to update within the migration window.** A specific two-hour window on a weekend afternoon works better than "sometime this week."
6. **Keep one repeater on the old preset temporarily.** Leave a repeater on the old preset for 1 - 2 weeks after migration so not-yet-migrated stragglers still have something to join. Note this does **not** bridge the two presets - a node on the old preset cannot hear or relay to nodes on the new preset (different modem settings mean no interoperability). It only continues serving users who haven't migrated yet, giving them a way to see a message like "Network has migrated to Medium Slow - update your preset to rejoin" and self-serve the fix. True bridging between two presets requires a dual-radio gateway running both presets at once.

Managing the Public Channel

You cannot control what strangers do on the public channel. Trolling, spam messages, and misconfigured nodes that flood the network with repeated traffic are occasional realities on any open mesh.

Mitigation strategies:

- Maintain a **secondary private channel** for your community's operational use, regardless of network size. The public channel is for newcomers and casual use; the private channel is for your community's serious coordination.
- Document and enforce a **community hop limit recommendation**. Node roles and rebroadcast settings can limit how infrastructure nodes relay traffic, which helps limit cascading retransmissions.
- Treat persistent disruptive nodes as a maintenance issue, not a personal conflict. Document the node ID and behavior, and discuss with your community whether any technical mitigations are warranted.

Tracking Growth Health

Watch these indicators month over month:

- Node count - use your own node database or the Meshtastic app for accurate total and online counts. Note meshmap.net only shows nodes that report to the public Meshtastic MQTT server, so a private community network (with MQTT uplink off) will not appear there in full and it is not a reliable node-count dashboard for your network.
- Channel utilization during peak hours
- Ratio of infrastructure repeaters to client nodes (more repeaters relative to total nodes = better network health)

Growth that outruns infrastructure is the most common failure mode for community meshes. Adding repeaters ahead of demand keeps the network healthy through growth phases.