Can I trust MeshCore encryption for sensitive communications?

MeshCore Encryption Summary

MeshCore provides two layers of encryption:

- **Channel encryption** - AES-128 ECB + HMAC-SHA256 with a key derived from the channel configuration. All nodes on the channel share this key. Note that public (no-PSK) channels and open hashtag rooms are readable by any passive listener.
- **Direct message encryption** - ECDH key exchange (Curve25519 elliptic curve) with AES session keys. Only sender and recipient can read DMs.

Cryptographic Strength

Both AES-128 and Curve25519 ECDH are modern, vetted cryptographic primitives used in TLS 1.3, Signal Protocol, and other high-security applications. Note, however, that those protocols use authenticated (AEAD) modes, whereas MeshCore's mode of use - AES-128 in ECB mode with a short 2-byte authentication tag and static ECDH keys - is weaker than the AEAD constructions in TLS 1.3 and Signal. MeshCore direct messages use sound primitives (X25519 ECDH key agreement + AES-128) but in a much simpler construction than Signal. Specifically, it uses AES-128 in ECB mode, a short 2-byte authentication tag, and static (long-term) keys - so it does NOT provide forward secrecy or the ratcheting that Signal uses. Treat it as basic confidentiality against casual interception, not as equivalent to Signal for high-sensitivity traffic.

What Encryption Protects Against

- **Passive eavesdroppers with LoRa hardware** - Cannot read message content with correct key management. (Public, no-PSK channels and open hashtag rooms remain readable by passive listeners.)
- **Casual interception of DM content** - Without the key, an eavesdropper cannot read DM plaintext. Note, however, that because DMs use AES-128 in ECB mode, identical plaintext blocks produce identical ciphertext, so an observer recording traffic long-term may detect repeated or structured content patterns (for example, templated emergency-comms messages) even without the key. Avoid sending fixed-format sensitive messages.
- **Replay/recorded-traffic decryption** - MeshCore's DM ECDH uses static long-term identity keys, not ephemeral session keys, so it does NOT provide forward secrecy. If a device's private key is later recovered, previously recorded DMs to or from that device can be decrypted. Replay resistance, where present, comes from per-message timestamps, not from the key exchange.

What Encryption Does NOT Protect Against

- **Physical access to your device** - Private keys are stored on the device. Seizure of the device potentially allows recovery of stored messages.
- **Compromised network operator** - A room server operator can see metadata (who is communicating with whom, when) and the content of room/channel posts that pass through the server; only person-to-person DMs are end-to-end encrypted and not readable by the operator.
- **Traffic analysis** - Observers can see that radio transmissions are occurring, timing patterns, and frequency of communication even if they cannot read content
- **Social engineering** - If a recipient shares a DM, encryption provides no protection
- **Channel key compromise** - If the shared channel key is obtained by an adversary, all past channel traffic (if recorded) can be decrypted
- **Loss of forward secrecy** - Because DM keys are static, recovering a private key compromises past as well as future messages; there is no per-message ratcheting like Signal's Double Ratchet.

Appropriate Use Cases

MeshCore's encryption is appropriate for:

- Community coordination that you'd prefer not to broadcast publicly
- Emergency operations where commercial networks are unavailable - but only as a supplemental, best-effort text channel. Mesh has no delivery guarantee, low throughput, and degrades under congestion; maintain a primary emergency-comms method and

never rely on mesh alone for life-safety traffic. The encryption above provides basic confidentiality only and is not suitable for life-safety or classified-equivalent secrecy.
- Private group communications within a trusted community

MeshCore encryption is *not* appropriate as the sole protection for:

- Communications subject to legal privilege (attorney-client, medical)
- Operational security against nation-state adversaries
- Highly sensitive personal information

For these use cases, use end-to-end encrypted applications (Signal, ProtonMail) with LoRa mesh serving only as a transport layer to reach an internet gateway.

Revision #7
Created 2026-05-03 05:53:12 UTC by Mesh America Admin
Updated 2026-06-08 19:53:33 UTC by Mesh America Admin