TitanMesh

Proof-of-Work Decentralized Messaging Protocol · No Coins, No Servers

Author: Hamed Rahmati

TitanMesh is a sovereign, peer-to-peer messaging layer that replaces cryptocurrency incentives with pure mathematics. Messages are secured by elliptic curve cryptography, protected from spam via client-side Proof-of-Work, and delivered through a global network of Raspberry Pi nodes. No tokens, no mining rewards—just a self-sustaining postal system running on CPU time and consensus.

End-to-End TitanMesh Protocol Flow

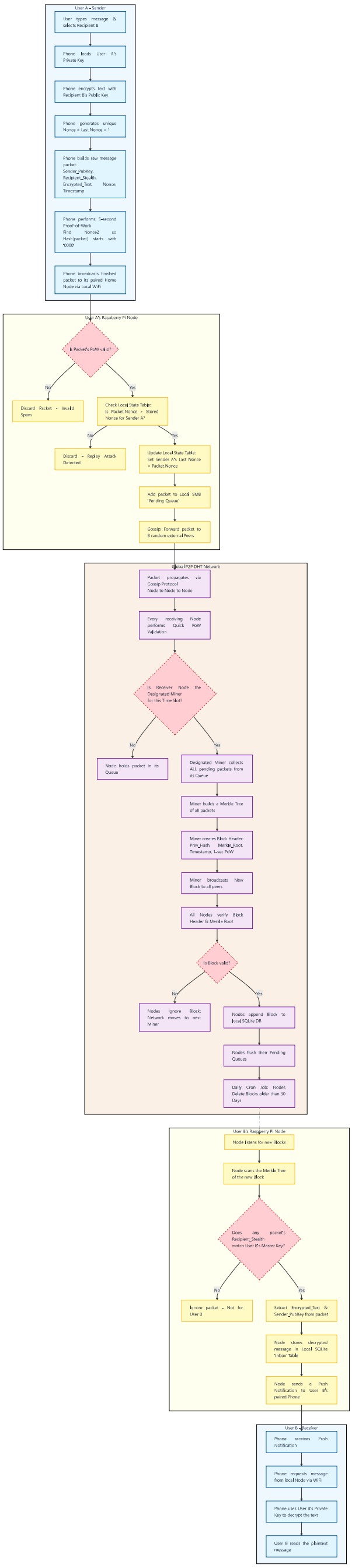

Figure 1: Complete TitanMesh message lifecycle — from encryption to pruning.

1. Core Philosophy

TitanMesh functions as a **global postal service running on mathematics** instead of financial incentives. There are no tokens, no wallets, and no miners competing for block rewards. Spam prevention relies on a simple physical constraint: **CPU time**. To send a single message, the sender's device must solve a 5-second Proof-of-Work puzzle. This makes flooding the network with millions of messages practically impossible—1 billion messages would require over 58 days of continuous computation.

Anybody can participate by running a node on a modest Raspberry Pi (64GB SD card, 2GB RAM, ~5 watts). The network is designed to be lightweight, censorship-resistant, and completely independent of centralized infrastructure.

2. Sending a Message (User A)

When Alice sends a message to Bob, her phone performs several cryptographic steps locally:

Encryption: The plaintext is encrypted with Bob's public key (Elliptic Curve Integrated Encryption Scheme).
Nonce Generation: A unique sequence number (previous nonce + 1) prevents replay attacks.
Proof-of-Work: The phone finds a nonce such that SHA-256(packet) begins with four zeros. This "postage stamp" takes roughly 5 seconds.
Stealth Addressing: Instead of exposing Bob's real public key, the packet contains a one-time stealth address derived from Bob's master key. Only Bob's node can recognize it.

The finished packet is sent over local Wi-Fi to Alice's own TitanMesh node (Raspberry Pi). It never touches a cloud server.

3. Sender's Node Validation

Alice's Raspberry Pi acts as her personal gatekeeper:

- **PoW Verification:** Checks the hash in under 0.01 seconds. Invalid packets are discarded instantly.
- **Replay Protection:** Compares the packet's nonce against the local state table. Only strictly increasing nonces are accepted.
- **State Update & Queuing:** The valid packet enters a small 5MB pending queue and is immediately gossiped to 8 random peers.

4. Global P2P Network & Consensus

The network is a DHT-based mesh of nodes communicating via Gossip Protocol. Every node quickly validates PoW before forwarding.

Designated Miner (No Mining Races): Instead of energy-wasting competition, the network deterministically selects one node per time slot (based on previous block hash and node count). This designated miner collects pending packets, builds a Merkle tree, and creates a block header with a trivial 1-second PoW.

The new block is broadcast, verified by all peers, and appended to a local SQLite database. Nodes then flush their pending queues. Every 24 hours, a cron job prunes blocks older than 30 days, keeping storage under 30GB.

5. Receiving & Decryption (User B)

Bob's node continuously scans new blocks. Using Bob's master private key, it checks each packet's stealth address. When a match is found, it extracts the encrypted payload and stores it in Bob's local inbox. A push notification is sent to Bob's phone over Wi-Fi.

Bob's phone retrieves the message, decrypts it with his private key, and displays the plaintext. Bob never sees the underlying PoW, Merkle trees, or block pruning.

6. Self-Defense Mechanisms

Threat	TitanMesh Countermeasure
Spam (1B messages)	5-second client PoW per message; 1B msgs = 58 CPU-days.
Replay Attacks	Strictly incrementing nonces tracked per sender.
Censorship	Gossip propagation; next designated miner includes pending packets.
Storage Bloat	Automatic 30-day pruning; archival nodes optional.

Threat	TitanMesh Countermeasure
Centralization	DHT peer discovery; no DNS seeds.
Metadata Leakage	Stealth addresses hide recipient; encryption hides content.

7. Raspberry Pi Node Specifications

CPU: ARM Cortex-A72 (Pi 4/5) - handles validation effortlessly.
RAM: 2 GB minimum (4 GB recommended).
Storage: 64 GB SD card (30 days of global messages).
Bandwidth: ~2 GB/day (less than one streaming movie).
Power: ~5 watts (~$5/year electricity).
Bootstrap: New node syncs block headers (~4 MB) in under 60 seconds.

8. Sovereign Communication Layer

TitanMesh delivers a permissionless, coinless messaging backbone. It requires no approval from Google, Apple, or any telecom. It runs on $50 hardware, secured by SHA-256, ECDSA, and the immutable cost of CPU cycles. This is **Bitmessage evolved—**optimized for the Raspberry Pi generation, free from mining farms and corporate gatekeepers.

TitanMesh Whitepaper v1.0

Hamed Rahmati · paxmatixamevt@gmail.com

"Pure math. Zero coins. Unstoppable messages."