## Abstract

Every system proposed in this research series — physical co-presence verification ((Applebee & Combe, 2026, "*Be In The Same Room*")), community emergency response ((Applebee & Combe, 2026, "*Wanted Attention for Unwanted Results*")), direct democracy voting ((Applebee & Combe, 2026, "*Swiss Direct Democracy*")), consent-based application distribution, and sovereign identity ((Applebee & Combe, 2026, "*Platform Gatekeeping*")) — requires a communication layer that operates without centralized infrastructure, without internet connectivity, and without permission from any corporation or government. This paper demonstrates that Bluetooth Low Energy (BLE) mesh networking, present in every smartphone manufactured since 2015, constitutes precisely this layer. We present the VexConnect protocol: a 7-hop, TTL-managed mesh with 1,000-entry deduplication cache, 480-byte chunked payload transfer, and cryptographically signed presence attestation via RSSI proximity bounds. We show that a network of ordinary smartphones running BLE 5.0+ can carry identity verification attestations, EIP-712 signed votes, emergency SOS broadcasts with ephemeral GPS, vouch ceremony signatures, and application payloads — all without Wi-Fi, cellular, or internet connectivity. The network has no server. No company runs it. No government can shut it down. The infrastructure already exists in 6.8 billion active smartphones worldwide. It is waiting to be turned on.

**Keywords:** BLE mesh, Bluetooth Low Energy, decentralized infrastructure, mesh networking, proximity attestation, civic technology, emergency response, offline voting, sovereign identity

## 1. Introduction

In 2021, Facebook went down for six hours. Government services in India, healthcare communication in Brazil, and emergency coordination across sub-Saharan Africa collapsed with it (BBC, 2021). Three billion people lost their primary communication infrastructure because a single company's Border Gateway Protocol configuration was misconfigured (Janardhan, 2021).

This is not a technology problem. It is a dependency problem. The world's communication infrastructure is routed through a small number of corporate chokepoints — and those chokepoints can fail, be disabled, or be deliberately shut down.

Governments have repeatedly demonstrated willingness to disable internet infrastructure during civil unrest. Between 2016 and 2023, internet shutdowns were documented in 74 countries, affecting an estimated 6.1 billion people and costing the global economy $53 billion (Access Now, 2024; Top10VPN, 2023). In each case, the ability to communicate, organize, verify identity, request emergency assistance, or participate in democratic processes was eliminated — not because the technology failed, but because the infrastructure was centrally controlled and the controller chose to turn it off.

Eduard Kaeser observes that "invisibility is a signum of power" (Kaeser, 2018, p. 28). Infrastructure becomes visible only when it fails — and that invisibility shields it from accountability. The Facebook outage made the dependency visible for six hours; then it was repaired and became invisible again. The BLE mesh inverts this: a network made of the people standing next to you is never invisible.

This paper presents an alternative: a communication layer that cannot be turned off because it has no center. Bluetooth Low Energy mesh networking uses the radios already present in smartphones to create ad-hoc networks that function without towers, without servers, without ISPs, and without corporate infrastructure of any kind.

The contribution of this paper is not the concept of BLE mesh networking, which is well established (Bluetooth SIG, 2017). The contribution is demonstrating that BLE mesh — with specific protocol extensions for identity attestation, vote relay, emergency broadcast, and consent-based payload transfer — constitutes sufficient infrastructure for the entire OMXUS civic technology stack. No additional hardware is required. No infrastructure investment is needed. The network is the people carrying their phones.

---

# 2. Technical Foundation

## 2.1 BLE 5.0+ Capabilities

Bluetooth Low Energy 5.0, ratified in 2016 and present in every smartphone manufactured since approximately 2017, provides:

| CAPABILITY | BLE 5.0 SPECIFICATION | |
|---|---|---|
| Range (outdoor, line of sight) | Up to 400m | |
| Range (indoor, typical) | 30-50m | |
| Data rate | 2 Mbps (LE 2M PHY) | |
| Advertising payload | 255 bytes (extended advertising) | |
| Connection-oriented data | Unlimited (via L2CAP) | |
| Power consumption | 10-20mA active, -85 | > 30 meters |

While RSSI is noisy and affected by obstacles, it provides a **hard upper bound on distance** that cannot be spoofed without specialized hardware. An RSSI of -50 dBm guarantees the transmitting device is within approximately 5 meters. This is sufficient for the co-presence protocol ((Applebee & Combe, 2026, "*Be In The Same Room*")), which requires proof that devices were in the same room, not their exact locations.

## 3.3 Vouch Attestation

The vouch ceremony (3-witness identity verification) produces a cryptographically signed attestation:

```
Attestation = {
  newHumanPubKey:    target's public key
  voucherTokenId:    voucher's HER token ID
  timestamp:         Unix epoch seconds
  bleRssi:           signal strength (proximity proof)
  nonce:             8 random bytes (replay prevention)
}

signature = secp256k1.sign(keccak256(attestation), voucher_private_key)
```

Verification:

1. Recover public key from signature
2. Derive Ethereum address from public key
3. Check isVerified(address) on HER contract
4. Confirm: only verified humans can vouch for new humans

**Anti-relay**: The RSSI value is included in the signed attestation. A relay attack would produce a valid signature but with an RSSI indicating the voucher was far away, which the verification contract can reject.

---

# 4. Emergency Broadcast

## 4.1 Design Principles

The emergency broadcast system (see also Papers 13, 14) operates on three principles:

1. Signed: Every emergency is cryptographically signed by the sender's identity key. False emergencies are attributable.
2. Ephemeral location: GPS coordinates are included for 60 seconds, then wiped. Responders get direction. Surveillance gets nothing.
3. Mesh-propagated: The SOS travels through the BLE mesh without internet. Every phone in range relays it.

## 4.2 Implementation

```
Emergency Packet = {
  type:        'EMERGENCY'
  alertId:     16 random bytes
```

```
  address:     sender's Ethereum address
  timestamp:   Unix epoch seconds
  lat, lon:    GPS coordinates (ephemeral)
  ttl:         5 (mesh hops)
}

signature = secp256k1.sign(keccak256(packet), sender_private_key)
```

After broadcast:

- GPS coordinates are zeroed after 60 seconds on the sending device
- Relay nodes forward but do not store GPS
- Responders within BLE range (~350m indoor, ~2.8km outdoor) receive the alert
- Vouch chain identifies the sender — responders know who needs help

### 4.3 Response Time

Current emergency response times (see (Applebee & Combe, 2026, "*Wanted Attention for Unwanted Results*") for full analysis):

| SYSTEM | AVERAGE RESPONSE |
|---|---|
| Police (Australia) | 15-45 minutes |
| Ambulance (Australia) | 12-18 minutes |
| Hatzolah (community) | 2-4 minutes |
| BLE mesh broadcast | (speed of light through relay chain) |

The mesh doesn't dispatch responders — it alerts every verified human within range simultaneously. The people closest respond first. No dispatcher. No routing. No wait.

# 5. Offline Voting

## 5.1 The Problem

Conventional electronic voting requires internet connectivity to submit votes to a central server or blockchain. This creates:

- Dependence on ISP infrastructure (can be shut down)
- Surveillance opportunities (who voted, when, from where)
- Exclusion of communities without reliable internet

## 5.2 The Solution: Mesh-Relayed EIP-712 Votes

Votes are signed locally using EIP-712 typed data (Ethereum Improvement Proposal 712), requiring zero gas and zero internet:

```
Vote = {
  proposalId:   uint256
  choice:       uint256 (0=ABSTAIN, 1=YES, 2=NO)
  voter:        address
  nonce:        uint256
}

digest = keccak256("\x19\x01" + domainSeparator + structHash(Vote))
signature = secp256k1.sign(digest, voter_private_key)
```

The signed vote is packaged as a mesh packet:

```
Mesh Vote Packet = {
  type: 'VOTE',
  proposalId, choice, voter, nonce,
```

```
  v, r, s  (signature components)
}
```

### 5.3 Relay Pipeline

```
VOTER (offline, no internet)
  ↓ Sign vote locally (zero gas, zero network)
  ↓ Package as VOTE mesh packet
  ↓ Broadcast via BLE (TTL=7)
  ↓
RELAY NODES (any phone in mesh range)
  ↓ Forward packet (don't need to understand content)
  ↓ Eventually reaches a node with internet
  ↓
ONLINE NODE
  ↓ Detects 'VOTE' packet type
  ↓ Calls VoteAussie.submitVoteBatch([signedVote]) on-chain
  ↓
BLOCKCHAIN
  → Vote recorded permanently. Verifiable. Immutable.
```

**Key property**: The vote is valid the moment it's signed. It doesn't matter if it reaches the blockchain in 5 seconds or 5 days. The cryptographic signature proves the voter's intent regardless of relay delay.

**Key privacy property**: Relay nodes see encrypted vote packets. They cannot determine who voted for what. Only the blockchain smart contract can verify the signature and record the vote.

---

## 6. Consent-Based Application Distribution

### 6.1 The Problem with App Stores

Application distribution is controlled by two companies: Apple and Google. Combined, they control access to software for 99%+ of smartphone users. Apps can be removed retroactively (Epic Games vs. Apple, 2020), blocked by geography (GitHub sanctions, 2019), or censored by content policy.

## 6.2 The Ripple: Peer-to-Peer App Distribution

VexConnect enables person-to-person application transfer via BLE mesh:

1. Advertise: Giver broadcasts PAYLOAD_AVAILABLE with size, hash, trust score
2. Consent: Receiver sees the offer and must explicitly accept — nothing downloads without tapping "Accept"
3. Transfer: 480-byte chunks via BLE, paced at 50ms intervals
4. Verify: SHA-256 integrity check on reassembly
5. Lineage: Generation counter embedded in HTML comment — tracks how many hands the app passed through

```
Lineage = {
  generation:   0 (origin) → 1 → 2 → 3...
  originHash:   SHA-256 of generation-0 payload
  spreadAt:     timestamp
  giverTrust:   trust score of person who gave it
  name:         optional plain-text identifier
}
```

## 6.3 Canary Ripple-Back

Each node periodically broadcasts a `CANARY` signal containing its generation number and origin hash. This lets the original creator see their application rippling outward through the network — concentric rings, one per generation — without knowing who has it or where they are.

**No surveillance. Just a heartbeat.** You released something into the world. The canary tells you it's alive.

# 7. Security Analysis

## 7.1 No Single Point of Failure

| ATTACK | CENTRALIZED NETWORK | BLE MESH |
|---|---|---|
| Server takedown | Network dies | No server to take down |
| ISP shutdown | Network dies | No ISP needed |
| DNS poisoning | Network misdirected | No DNS |
| Government internet kill switch | Network dies | Unaffected |
| Corporate deplatforming | Users locked out | No platform to deplatform from |
| DDoS | Server overwhelmed | No server to DDoS |

## 7.2 Attack Vectors and Mitigations

**Jamming**: BLE operates in the 2.4GHz ISM band. Jamming is possible but:

- Illegal in most jurisdictions
- Requires physical proximity to the target area
- Detectable by spectrum analysis
- BLE frequency hopping (79 channels) provides inherent resilience
- Jamming one area doesn't affect the rest of the mesh

**Replay attacks**: Prevented by packet ID (SHA-256 of nonce + payload) and 1,000-entry deduplication cache with 60-second TTL.

**Sybil attacks on the mesh**: Node IDs are ephemeral. Creating many fake nodes doesn't help because the mesh carries signed payloads — the identity system ((Applebee & Combe, 2026, "*Be In The Same Room*")) handles

Sybil resistance at the identity layer, not the network layer.

**Eavesdropping**: BLE packets are broadcast and can be received by any device in range. Mitigations:

- Vote packets contain signatures but votes can be encrypted
- Emergency packets intentionally broadcast (that's the point)
- Vouch attestations are signed and public by design
- Payload transfers use SHA-256 integrity verification

## 7.3 Graceful Degradation

The mesh degrades gracefully as node density decreases:

| NODES PER KM² | COVERAGE | CAPABILITY |
|---|---|---|
| > 100 | Continuous | Full mesh: voting, emergency, vouch, payload |
| 50-100 | Intermittent | Store-and-forward: votes relay when density increases |
| 10-50 | Sparse | Local only: vouch ceremony (4 people), emergency (direct) |