GhostWire Security & Privacy

Contents

Security & Privacy

Executive Summary

GhostWire is built with security and privacy as core principles. This chapter explains how GhostWire protects users, what threats it defends against, and how to use its security features—whether you're a non-technical user or a security engineer.

Security Foundations (Plain & Technical)

- **Plain:** GhostWire keeps your messages private and your identity safe, even if someone tries to spy or block you.
- **Technical:** End-to-end encryption (AES-256-GCM, X25519), perfect forward secrecy, ephemeral keys, key rotation, secure storage, and post-quantum crypto (planned).

Threat Model (Real-World)

- **Adversaries:** Censors, eavesdroppers, Sybil attackers (fake nodes), spammers, malicious insiders.
- **Scenarios:**
  - Protesters in a censored country use GhostWire to avoid surveillance.
  - Disaster responders rely on GhostWire to keep messages private when infrastructure is down.
  - Rural users defend against spam and fake nodes with Sybil defense and quotas.

Key Security Features

- **End-to-End Encryption:** All messages are encrypted in transit and at rest.
- **Perfect Forward Secrecy:** Session keys rotate, so past messages stay safe even if a key is compromised.
- **Ephemeral Identities:** Temporary keys for privacy.
- **Sybil Defense:** Prevents fake node attacks using reputation, proof-of-work, and social trust.
- **Quotas & Blacklists:** Rate limits and blocklists to stop spam and abuse.
- **Disaster Triggers:** Emergency wipe, panic button, rapid shutdown.
- **Reputation & Federation:** Trust is earned, bad actors are penalized, and networks can interconnect with trust boundaries.
- **Traffic Obfuscation:** Cover traffic, timing randomization, and packet padding to resist surveillance.

Best Practices (Checklist)

- Always use the latest release.
- Rotate keys regularly.
- Enable all security modules.
- Report vulnerabilities via GitHub Security.
- Use strong passwords and device security.

Anti-Patterns (What Can Go Wrong)

- Using outdated software.
- Disabling encryption or security modules.
- Sharing devices or credentials.
- Ignoring suspicious activity or failing to report bugs.

Technical Deep Dive

- **Key Management:** Ed25519/X25519, ephemeral keys, secure storage, automated/manual rotation.
- **Trust Modules:** Sybil defense, quotas, blacklists, disaster triggers, reputation, federation, traffic obfuscation.
- **Threat Mitigations:** Strong crypto, modular trust, rapid response to threats.

Real-World Security Scenarios

- **Censorship Resistance:** Users in hostile environments use Stealth TCP and traffic obfuscation to avoid detection.
- **Disaster Mode:** Emergency responders trigger rapid shutdown and data wipe if devices are at risk.
- **Community Trust:** Nodes build reputation over time, and bad actors are automatically penalized.

Further Reading & Resources

- AWS Well-Architected Security Pillar
- OWASP Top 10
- NIST Cybersecurity Framework

GhostWire: Security by design, privacy by default.