Appendix D – Threat Scenarios & Attack Trees

Author: Ahmad Hemmati | Website: https://www.twincodesworld.com

Status: Open Source – Technical Supplement | Version: v1.0

This appendix explains the Threat Models and Attack Trees for LHDNS to determine what type of attack it is designed against.

1. Threat Categories

- **Passive Surveillance**
  - *Threat:* Adversaries monitoring traffic to deanonymize users.
  - *Mitigation:* Onion routing, cover traffic, short TTLs, ephemeral tokens.
- **Active Censorship**
  - *Threat:* State-level actors blocking resolution requests or injecting false responses.
  - *Mitigation:* Ledger consensus, multi-path routing, gossip propagation, cryptographic verification.
- **Sybil Attacks**
  - *Threat:* Adversary floods the network with fake nodes to gain majority influence.
  - *Mitigation:* Staking requirements, slashing misbehavior, reputation systems.
- **Eclipse Attacks**
  - *Threat:* A node's connections are monopolized by malicious peers.
  - *Mitigation:* Random peer sampling, peer diversity checks, rotating relay sets.
- **Replay & Injection Attacks**
  - *Threat:* Adversary replays old tokens or injects fake descriptors.
  - *Mitigation:* Nonce-based ephemeral tokens, signature validation, strict TTL expiry.
- **DoS / DDoS Attacks**
  - *Threat:* Flooding the network with bogus resolution requests.
  - *Mitigation:* Rate limiting, micropayment requirements, proof-of-work throttling.
- **Traffic Correlation**
  - *Threat:* Linking entry and exit traffic patterns to deanonymize users.
  - *Mitigation:* Multipath relays, cover traffic, randomized delays.
- **Key Compromise**
  - *Threat:* Long-term keys are stolen.
  - *Mitigation:* Ephemeral key exchange, forward secrecy, rapid key rotation.

2. Attack Tree Example: Deanonymization

Goal: Identify user-service mapping

- Level 1: Observe traffic
    - Level 2a: Monitor entry nodes
    - Level 2b: Monitor exit relays
- Level 1: Correlate flows
    - Level 2a: Perform timing analysis
    - Level 2b: Perform volume analysis
- Level 1: Replay tokens
    - Level 2a: Inject old hash-token into network
    - Level 2b: Force service to respond

Countermeasures: Cover traffic, randomized delays, ephemeral hash-tokens, replay protection.

3. Attack Tree Example: Service Takedown

Goal: Prevent a service from being reachable

- Level 1: Block ledger entries
    - Level 2a: Jam gossip propagation
    - Level 2b: Fork ledger
- Level 1: Overload service descriptor updates
    - Level 2a: Flood with fake descriptors
    - Level 2b: Exploit long-lived entries
- Level 1: Isolate service node
    - Level 2a: Eclipse attack
    - Level 2b: BGP hijack (legacy interop)

Countermeasures: Gossip resilience, multi-path propagation, signature validation, short TTLs, compliance gateways fallback.

4. Summary

LHDNS's layered defense-in-depth design anticipates both **passive surveillance** and **active adversaries**, including state-level attackers. By combining **ledger accountability, cryptographic guarantees, and privacy-preserving routing**, it minimizes attack surfaces while keeping the system functional and scalable.