LLM-Tor: Unlinkable Access to Commercial LLM APIs via Blind Signatures and Tor

Prince Gupta

February 17, 2026

Abstract

LLM-Tor is a cryptographically enforced privacy layer that enables unlinkable access to commercial Large Language Model (LLM) APIs. It separates payment identity from model usage using RSA blind signatures (RFC 9474) and onion-routed communication via Tor. The system prevents correlation between user identity and chat content at the proxy layer while maintaining compatibility with frontier commercial LLM providers.

Contents

1 Introduction

Commercial LLM APIs require authenticated access tied to user accounts. This creates a structural privacy issue: prompts and responses become permanently associated with identifiable users.

LLM-Tor introduces a cryptographic separation between:

- Payment identity
- Usage authorization
- Model inference requests

The goal is to ensure that even the proxy operator cannot link a specific prompt to a specific paying customer.

2 Background

2.1 Blind Signatures

LLM-Tor uses RSA Blind Signatures as defined in RFC 9474.

A blind signature scheme allows a signer to sign a message without learning its contents. The protocol consists of:

1. Blinding
2. Signing
3. Unblinding
4. Verification

Security relies on the hardness of RSA and the one-more RSA assumption.

Let:

$$m = \text{random token}$$

Client computes:

$$m' = m \cdot r^e \mod N$$

Server signs:

$$s' = (m')^d \mod N$$

Client unblinds:

$$s = s' \cdot r^{-1} \mod N$$

The resulting signature s is valid for m, but the signer never sees m.

2.2 Tor Onion Routing

Tor provides layered encryption and multi-hop routing to hide client IP addresses. LLM-Tor exposes its inference endpoint to be accessed via tor exit nodes. In future it will be exclusively accessible as a Tor onion service.

Reasons for using Tor:

- Prevent IP-level linkage during token redemption

- Eliminate DNS metadata leakage
- Provide probabilistic anonymity against network observers

LLM-Tor does not rely on Tor for cryptographic unlinkability, but uses it to prevent network-layer identity correlation.

3 System Architecture

3.1 Components

- Payment Service
- Credit Ledger
- Blind Signing Service
- Token Redemption Proxy
- Moderation Layer
- Upstream LLM Provider

3.2 Protocol Phases

3.2.1 Phase 1: Credit Purchase

1. User purchases credits via standard payment provider.
2. Server records credit balance linked to user identity.

Identity exists in this phase.

3.2.2 Phase 2: Blind Token Issuance

1. Client generates random token T.
2. Client blinds T using model-specific public key.
3. Server verifies credit availability.
4. Server blind-signs the blinded token.
5. Client unblinds signature.

The server never observes T in unblinded form. Identity exists in this phase.

3.2.3 Phase 3: Anonymous Redemption

1. Client establishes Tor circuit.
2. Client submits (T, σ).
3. Server verifies RSA signature.
4. Server checks token not previously spent.
5. Server forwards prompt to upstream LLM.

4 Token Structure

Each token contains:

- Random 128-bit value
- Model identifier: implicitly linked by choosing public key specific for that model.
- Key identifier (for rotation) [Will be done in future]

Spent tokens are stored as hash(token) to prevent replay.

5 Security Properties

5.1 Unlinkability

Under the security assumptions of RSA blind signatures, issued tokens cannot be linked to redeemed tokens.

5.2 Single-Use Enforcement

Tokens are marked spent atomically upon first successful verification. Replay attempts are rejected.

5.3 Identity Separation

Payment identity never appears in redemption requests. Redemption occurs exclusively via Tor onion service.

6 Moderation Requirement

LLM-Tor performs content moderation prior to forwarding requests upstream.

Reasons:

- Legal compliance in certain jurisdictions
- Abuse mitigation (spam, illegal content)
- Prevention of infrastructure misuse

Moderation is performed with the content at which point it's not attachable to the user identity.

7 Threat Model

7.1 Adversaries

- Honest-but-curious proxy operator
- External network observer
- Malicious user attempting double-spend
- Upstream LLM provider

7.2 Security Goals

- Prevent proxy from linking prompt to payment identity
- Prevent token double spending
- Prevent IP-level identity leakage

7.3 Out of Scope

- Global passive adversary controlling Tor
- Endpoint compromise
- Logging by upstream LLM providers
- Stylometric fingerprinting

8 Key Rotation Policy

TODO

9 Limitations

LLM-Tor does not provide absolute anonymity. It provides unlinkability at the proxy layer under standard cryptographic assumptions with the added benefit of network anonymity with tor directly built in the official client.

10 Future Work

- Exclusive tor onion hosting for proxy microservice
- Key Rotation
- Zero-knowledge token systems
- Anonymous payment integration
- Post-quantum blind signatures
- Decentralized moderation

11 Conclusion

LLM-Tor demonstrates that cryptographic blind signatures combined with onion routing can provide practical unlinkability for commercial LLM API access, without requiring local model hosting.