AIMB-X Whitepaper

Cryptographically Verified AI Inference Blockchain

Public Testnet Edition

Abstract

AIMB-X is an AI-native and Zero-Knowledge (ZK)-native Layer-1 blockchain designed to make **AI inference cryptographically verifiable**, not merely observable or trusted. Unlike conventional AI-on-chain approaches that rely on trusted execution environments, off-chain attestations, or operator reputation, AIMB-X treats *AI outputs as untrusted by default* and enforces correctness through zero-knowledge proofs bound directly to on-chain execution.

This whitepaper describes the motivation, architecture, protocol design, economic model, and public testnet implementation of AIMB-X.

---

1. Motivation

1.1 The Trust Problem in AI Systems

Modern AI systems are increasingly embedded in high-stakes workflows:

- Financial execution
- Governance automation
- Autonomous agents
- Smart contract decision logic

However, most systems implicitly assume:

If an AI ran on reputable infrastructure, its output can be trusted.

This assumption breaks under adversarial conditions, economic incentives, and decentralized environments.

1.2 Why On-Chain AI Is Not Enough

Putting AI *on-chain* without verifiability only shifts trust:

- From model correctness → operator correctness
- From centralized servers → validators

AIMB-X rejects trust aggregation and instead enforces **cryptographic correctness**.

---

2. Core Principle

Do not trust AI outputs. Verify them cryptographically.

Every AI inference that influences protocol state must be accompanied by a **zero-knowledge proof** that:

- The model was executed faithfully
- The inputs were bound to on-chain context
- The output was not altered or forged

---

3. System Overview

AIMB-X is composed of the following layers:

User / Agent
↓
AI Inference Request
↓
ZK-Inference Circuit
↓
Proof Verification
↓
VM State Transition
↓
Blockchain Finality

---

4. ZK-Inference Architecture

4.1 ZK Inference Circuit

The ZK circuit proves:

- Model weights commitment
- Input tensor commitment
- Deterministic execution trace
- Output tensor correctness

No raw model parameters or private inputs are revealed.

4.2 Context Binding

Each inference proof is bound to:

- Block height
- Chain ID
- VM execution context
- Caller address

This prevents replay, reuse, or cross-context attacks.

---

5. AIMB-X Virtual Machine (VM)

The AIMB-X VM is designed to:

- Treat AI inference as a **first-class opcode**
- Enforce proof verification before state mutation
- Reject unverifiable outputs deterministically

AI execution is not an oracle; it is a **provable computation**.

---

6. Consensus & Execution Model

- Deterministic block production
- Proof-first execution ordering
- Stateless verification of AI correctness

Validators do **not** re-run AI models. They verify proofs only.

---

7. Token Model (Testnet)

7.1 Testnet Token

- Non-economic
- Used for:
    - Transaction fees
    - Spam prevention
    - Execution accounting

7.2 Faucet-Based Distribution

- New wallets start with **0 balance**
- Tokens are obtained via explicit faucet claim
- No implicit or auto-funded accounts

This enforces realistic economic assumptions from day one.

---

8. Wallet & Developer Experience

8.1 Wallet Design

- Deterministic key generation
- Explicit balance query via RPC
- Offline signing support

8.2 Web Wallet & Explorer

Public testnet includes:

- Web wallet UI
- Block explorer
- Proof indicators
- State root visibility

All interfaces display **verification status**, not trust labels.

---

9. Security Model

AIMB-X defends against:

- Malicious model execution
- Forged AI outputs
- Replay attacks
- Context substitution

Assumptions:

- ZK proof system security
- Cryptographic hash collision resistance

No assumptions are made about operator honesty.

---

10. Public Testnet Goals

The public testnet is designed to:

- Validate protocol architecture
- Test developer onboarding
- Stress adversarial AI scenarios
- Collect feedback on proof latency and UX

No mainnet guarantees are implied.

---

11. Roadmap

- **Stage S**: Developer Tooling & SDKs
- **Stage T**: Adversarial Inference Campaigns
- **Stage U**: Governance Simulation
- **Stage V**: Economic Testnet

---

12. Conclusion

AIMB-X introduces a new paradigm:

AI systems that are not *trusted*, but *provably correct*.

By making cryptographic verification a first-class requirement, AIMB-X enables AI to safely participate in decentralized, adversarial, and economically sensitive environments.
