Lux Z-Chain: Privacy-Preserving Smart Contracts with Zero-Knowledge Proofs

Lux Partners
research@lux.network

October 2025

Abstract

We present **Lux Z-Chain**, a privacy-focused Layer-2 subnet enabling confidential smart contracts via zero-knowledge proofs. Z-Chain combines **zk-SNARKs** for transaction privacy, **FHE (Fully Homomorphic Encryption)** for encrypted computation, and **TEE (Trusted Execution Environments)** for hybrid privacy guarantees. Key contributions: (i) zk-EVM with **100+ TPS** throughput for private transactions, (ii) Confidential token standard (LRC-721P) compatible with NFTs and DeFi, (iii) Privacy-preserving DeFi primitives (swap, lend, stake) with **¡ 2s finality**, (iv) Regulatory compliance via selective disclosure and auditor keys. Deployed on testnet, Z-Chain has processed **1.2M private transactions** with **zero privacy breaches**.

1 Introduction

Public blockchains expose all transaction data on-chain, creating privacy challenges for:

- **Individuals**: Wallet balances and transaction history publicly visible
- **Enterprises**: Business logic and trading strategies exposed
- **Institutions**: Regulatory compliance requires selective disclosure, not full transparency

Prior Work. Existing privacy solutions have limitations:

- **Zcash**: Privacy only for native token, no smart contracts
- **Monero**: Ring signatures have limited anonymity set
- **Aztec**: zk-Rollup has 30+ second proof generation
- **Tornado Cash**: Mixer contracts sanctioned by OFAC

Our Solution. Lux Z-Chain provides **programmable privacy** via zk-EVM, enabling private smart contracts with DeFi composability and regulatory compliance.

2 Architecture

2.1 Privacy Model

Z-Chain offers **three privacy tiers**:

Users select privacy tier per transaction based on requirements.

Tier	Privacy Level	Technology	Use Case
Tier 0	Public	Standard EVM	Transparent DeFi
Tier 1	Shielded	zk-SNARKs	Private transfers
Tier 2	Confidential	FHE	Encrypted DeFi
Tier 3	Trusted	TEE (SGX/SEV)	Regulated finance

Table 1: Z-Chain privacy tiers and technology stack

2.2 System Components

- **zk-EVM**: Zero-knowledge virtual machine for private smart contracts
- **Proof Generators**: Distributed provers generating zk-SNARKs
- **FHE Coprocessor**: Encrypted computation for Tier 2 contracts
- **TEE Validators**: SGX/SEV enclaves for Tier 3 contracts
- **Auditor Registry**: Authorized auditors with selective disclosure keys

3 zk-EVM Architecture

3.1 zkEVM Design

Z-Chain implements a **Type-3 zkEVM** (EVM-equivalent bytecode):

1. User submits shielded transaction T
2. Sequencer executes T off-chain, generates witness w
3. Prover generates zk-SNARK proof π:

$$\pi \leftarrow \text{Prove}\left(\text{ValidExec}(T, w, \text{state}_{\text{old}}, \text{state}_{\text{new}})\right) \tag{1}$$

4. L1 verifier checks π and updates state commitment

Privacy Guarantee: L1 sees only state commitment $C = \text{Hash}(\text{state}_{\text{new}})$, not transaction details.

3.2 Proof System

Circuit Constraints:

- EVM opcode execution: 2.1M constraints
- Merkle proof verification: 850k constraints
- Signature verification (ECDSA): 1.5M constraints
- Total: **4.45M constraints**

Performance:

- Proof generation: 6.8s per transaction
- Proof size: 288 bytes (Groth16)
- Verification time: 12ms on-chain
- Gas cost: 280k per proof

4 Confidential Token Standard

4.1 LRC-721P (Private NFTs)

Extension of ERC-721 with privacy:

Algorithm 1 Shielded NFT Transfer

Input: NFT ID n, recipient address A_{recv}, nullifier ν
Output: zk-SNARK proof π

// Prove ownership without revealing identity
$\text{commitment}_{\text{old}} \leftarrow \text{Hash}(n, A_{\text{sender}}, \text{salt})$
$\text{commitment}_{\text{new}} \leftarrow \text{Hash}(n, A_{\text{recv}}, \text{salt}')$

// Public inputs: $(\text{commitment}_{\text{old}}, \text{commitment}_{\text{new}}, \nu)$
// Private inputs: $(n, A_{\text{sender}}, A_{\text{recv}}, \text{salt}, \text{salt}')$

$$\pi \leftarrow \text{Prove}\begin{pmatrix} \text{commitment}_{\text{old}} \text{ in Merkle tree} \\ \wedge\ \nu = \text{Hash}(n, A_{\text{sender}}) \\ \wedge\ \text{commitment}_{\text{new}} = \text{Hash}(n, A_{\text{recv}}, \text{salt}') \end{pmatrix}$$

return π

Privacy Properties:

- NFT ownership hidden (only commitment visible)
- Transfer recipient hidden (encrypted address)
- Transfer history unlinkable (nullifiers prevent double-spend)
- Optional metadata disclosure via auditor key

5 Privacy-Preserving DeFi

5.1 Shielded DEX

Private Token Swap Protocol:

1. User deposits tokens A into shielded pool (generates commitment C_A)
2. User submits swap order $(C_A, B_{\text{amount}}, \text{price})$ via zkSNARK
3. DEX matches orders off-chain

4. User withdraws tokens B via proof π_B:

$$\pi_B \leftarrow \text{Prove}\left(\text{OwnsCommitment}(C_A) \wedge \text{ValidSwap}(A \rightarrow B)\right) \tag{2}$$

Advantages:

- Order book hidden (prevents front-running)
- Trading volume private (hides whale activity)
- Slippage protected (encrypted order matching)

5.2 Private Lending

Confidential Loan Protocol:

Action	Privacy Level
Collateral deposit	Shielded (zk-SNARK)
Loan amount	Encrypted (FHE)
Interest rate	Public (on-chain)
Liquidation threshold	Encrypted (FHE)

Table 2: Privacy levels in Z-Chain lending protocol

Key Feature: Liquidations occur via encrypted threshold checks (FHE-based), preserving collateral privacy until liquidation event.

6 Fully Homomorphic Encryption (FHE)

6.1 FHE Integration

For Tier 2 contracts, Z-Chain uses **TFHE (Threshold FHE)**:

- **Encryption**: User encrypts inputs under FHE public key
- **Computation**: Smart contract operates on ciphertexts
- **Decryption**: Threshold decryption by validator committee

6.2 Supported Operations

Operation	Gas Cost	Latency
Addition	50k	0.1ms
Multiplication	250k	2ms
Comparison ($<, >$)	180k	1.5ms
AND/OR/XOR	40k	0.08ms

Table 3: FHE operation costs and performance

Example Use Cases:

- Encrypted auctions (bids hidden until reveal)
- Private voting (encrypted vote tallying)
- Confidential credit scores (encrypted FICO-like computation)

7 Trusted Execution Environments (TEE)

7.1 Tier 3 Privacy Model

For regulated use cases, Z-Chain supports **TEE-based privacy**:

- Validators run Intel SGX or AMD SEV enclaves
- Smart contracts execute inside secure enclave
- Auditors receive encrypted attestations from TEE
- Regulators access transaction data via auditor keys

7.2 Attestation Protocol

Algorithm 2 TEE Transaction Attestation

Input: Transaction T, auditor public key pk_{aud}
Output: Encrypted attestation E, TEE quote Q

// Execute transaction in enclave
result $\leftarrow$ ExecuteInEnclave(T)

// Generate attestation
$A \leftarrow \{\text{sender}, \text{recipient}, \text{amount}, \text{timestamp}\}$
$E \leftarrow \text{Encrypt}(A, pk_{\text{aud}})$ ▷ Auditor can decrypt

// Remote attestation quote
$Q \leftarrow$ GenerateQuote(enclave_measurement)
return (E, Q)

Compliance Guarantee: Regulators verify TEE quote Q proves correct enclave execution, then decrypt E to audit transaction.

8 Selective Disclosure

8.1 Auditor Key System

Z-Chain implements **hierarchical auditor keys**:

1. **User Keys**: Can view own transaction history
2. **Contract Auditor Keys**: Can view all contract transactions

3. **Regulatory Keys**: Can view transactions matching specific criteria (e.g., > $10k transfers)
4. **Court Order Keys**: Can view specific addresses (requires on-chain governance vote)

8.2 View Key Protocol

Generating View Key:

$$vk = \text{HKDF}(sk_{\text{user}}, \text{"view\_key"}, \text{salt}) \tag{3}$$

Decrypting Commitment:

$$\text{PlaintextData} = \text{Decrypt}(C, vk) \tag{4}$$

Auditors receive vk (not sk_{user}), enabling read-only access without spending authority.

9 Security Analysis

9.1 Privacy Guarantees

[Transaction Privacy] Under the DDH assumption and random oracle model, an adversary viewing only $L1$ commitments cannot distinguish between two transactions with different amounts/recipients with advantage greater than $\text{negl}(\lambda)$.

Proof Sketch: Commitments are computationally hiding under DDH. zkSNARK zero-knowledge property ensures proofs leak no information beyond validity. □

9.2 Anonymity Set Size

Shielded Pool Size (as of Q4 2024):

- Total commitments: 1.2M
- Daily active commitments: 15k
- Effective anonymity set: $\approx 10^5$ per transaction

Compared to:

- Zcash shielded pool: 2.8M (but only 15% adoption)
- Monero ring size: 16 (small anonymity set)
- Tornado Cash: 50k (pre-sanctions)

10 Performance Evaluation

10.1 Throughput Benchmarks

10.2 Proof Generation Latency

11 Regulatory Compliance

11.1 AML/KYC Integration

Z-Chain supports **compliance without compromising user privacy**:

Transaction Type	TPS	Finality	Cost
Public (Tier 0)	5,000	1.5s	\$0.001
Shielded (Tier 1)	120	1.8s	\$0.08
FHE (Tier 2)	50	2.2s	\$0.15
TEE (Tier 3)	200	1.6s	\$0.02

Table 4: Z-Chain throughput by privacy tier

Circuit	Constraints	Prove Time	Proof Size
Transfer	280k	1.2s	288 bytes
Swap	850k	3.5s	288 bytes
NFT mint	420k	1.8s	288 bytes
Loan borrow	1.2M	5.1s	288 bytes

Table 5: zk-SNARK proof generation performance

1. User completes KYC with licensed provider (off-chain)
2. Provider issues **compliance certificate** (zk-attestation)
3. User submits certificate with shielded transaction
4. Smart contract verifies certificate without learning user identity

Certificate Proof:

$$\pi_{\text{kyc}} \leftarrow \text{Prove}\left(\text{HasValidCertificate}(pk_{\text{user}}, \text{provider}_{\text{id}})\right) \tag{5}$$

11.2 OFAC Compliance

To prevent sanctioned addresses, Z-Chain implements **nullifier blacklist**:

- Regulators submit sanctioned nullifiers to on-chain registry
- Smart contracts reject transactions with blacklisted nullifiers
- Privacy preserved: Only nullifier visible, not user identity

Key Advantage: Compliance without address-level deanonymization.

12 Deployment

12.1 Testnet Metrics

Z-Chain Testnet (Q3-Q4 2024):

- Transactions processed: 1.2M
- Unique addresses: 45k
- Shielded pool TVL: \$18M (testnet tokens)

- Average finality: 1.85s
- Privacy breaches: 0

12.2 Mainnet Roadmap

Phase	Timeline
Testnet v1 (zk-SNARKs only)	Q3 2024
Testnet v2 (+ FHE)	Q4 2024
Audit (Trail of Bits + OpenZeppelin)	Q1 2025
Mainnet launch (Tier 0-1)	Q2 2025
FHE mainnet (Tier 2)	Q3 2025
TEE mainnet (Tier 3)	Q4 2025

Table 6: Z-Chain deployment roadmap

13 Future Work

13.1 Post-Quantum zk-SNARKs

Transitioning to quantum-resistant proof systems:

- zk-STARKs (no trusted setup, but 100× larger proofs)
- Lattice-based zkSNARKs (research phase)
- Hybrid SNARKs + STARKs (practical quantum resistance)

13.2 Cross-Chain Privacy

Enabling private transfers across chains:

- Shielded bridge with Lux L1/L2
- IBC privacy module for Cosmos
- Private cross-rollup communication

14 Conclusion

Lux Z-Chain provides **programmable privacy** for smart contracts via zk-SNARKs, FHE, and TEEs. With **120 TPS for shielded transactions** and **¡ 2s finality**, Z-Chain enables privacy-preserving DeFi with regulatory compliance. Testnet deployment with **1.2M transactions and zero breaches** demonstrates the viability of practical blockchain privacy.

A Cryptographic Primitives

A.1 zk-SNARK Parameters

Groth16 Setup:

- Trusted setup ceremony: 256 participants
- Powers of tau: 2^{22} (4.2M constraints)
- Proving key: 1.8 GB
- Verification key: 3.2 KB

A.2 FHE Parameters

TFHE Configuration:

- Security level: 128-bit (post-quantum)
- Ciphertext size: 8 KB per encrypted integer
- Bootstrap time: 15ms
- Threshold: $t = 2/3$ of validators required for decryption

B Solidity Interfaces

```
interface IZChainPrivacy {
  // Deposit into shielded pool
  function deposit(uint256 amount, bytes32 commitment)
    external returns (bool);

  // Shielded transfer (requires zk-SNARK proof)
  function transfer(bytes32 nullifier, bytes32 newCommitment,
    bytes calldata zkProof) external returns (bool);

  // Withdraw from shielded pool
  function withdraw(uint256 amount, bytes calldata zkProof,
    address recipient) external returns (bool);
}
```

Disclaimer. This document describes a testnet protocol. Mainnet security guarantees depend on successful audits and cryptographic assumptions.