vBitZK

The Beneficial Ownership Standard for the Tokenized Economy

Zero-Knowledge Proofs of Ultimate Control — v1.1 (December 2025)

312 bytes. 180 ms. $0.001. Trustless. The compliance primitive for $100 T on-chain wealth.

Executive Summary

By 2030, Boston Consulting Group, BlackRock, and McKinsey project **$16–30 trillion** of real-world assets will live on-chain. Every single one of those assets must answer the same question regulators have asked since 2008:

> *Who is the ultimate beneficial owner and what is their exact economic exposure?*

Today the answer is **"nobody knows"** once assets enter DeFi. vBitZK v1.1 changes that forever.

A **312-byte zero-knowledge proof** now verifiably answers that question through **32 layers** of DeFi nesting in under **180 ms** for **$0.001**, fully decentralized, on-chain verifiable in **62k gas** on eight chains.

This is not analytics. This is the compliance rail the entire tokenized economy will run on.

Competitive Landscape

Feature	vBitZK v1.1	Chainalysis KYT	TRM Labs	Elliptic	Arkham	Nansen
Recursive DeFi unwrapping	**32 layers**	1–2 layers	1 layer	1 layer	1 layer	1 layer
Zero-knowledge proof	**Yes**	No	No	No	No	No
Proof size	**312 bytes**	N/A	N/A	N/A	N/A	N/A
Proving time	**<180 ms**	N/A	N/A	N/A	N/A	N/A
Cost per proof	**$0.001**	N/A	N/A	N/A	N/A	N/A
On-chain verification	**62k gas**	N/A	N/A	N/A	N/A	N/A
Decentralized proving	**Cysic**	Centralized	Centralized	Centralized	Centralized	Centralized
SAR/CTR auto-generation	**Built-in**	Manual	Manual	Manual	Manual	Manual
Privacy-preserving	**Yes**	No	No	No	No	No
Cross-chain proofs	**8 chains**	Multi-chain	Multi-chain	Multi-chain	Limited	Limited

Bottom line: Analytics platforms tell you *what happened*. vBitZK proves *who owns what* — cryptographically, privately, on-chain.

What vBitZK Does

1. Resolves Beneficial Ownership Through DeFi Layers

```
Wallet: 0xd8dA6BF26964aF9D7eEd9e03E53415D37aA96045
    │
    ├── 100 weETH (Ether.fi)
    │       └── 118 eETH (rebasing)
    │               └── 118 stETH (Lido)
    │                       └── 118 ETH ← terminal
    │
    ├── 50,000 PT-stETH (Pendle)
    │       └── 47,500 stETH (at maturity)
    │               └── 47,500 ETH ← terminal
    │
    └── 25,000 aUSDC (Aave V3)
            └── 25,000 USDC ← terminal

═══════════════════════════════════════════════════

BENEFICIAL OWNERSHIP RESOLVED:
├── 165.5 ETH exposure (79.2%)
└── 25,000 USDC exposure (20.8%)
═══════════════════════════════════════════════════
```

2. Generates Zero-Knowledge Proofs

The proof cryptographically commits to:

Commitment	Description
WHO	Root wallet address controlling assets
WHAT	Terminal assets (ETH, USDC, WBTC, etc.)
HOW MUCH	Exact economic exposure in basis points
WHEN	Proof expiration (90 days default)
VERIFIED BY	KYC hash without revealing identity

Anyone can verify on-chain. **No one learns the identity.**

3. Enables Regulatory Compliance

Machine-readable compliance reports generated automatically:

- **SAR (Suspicious Activity Report)** — FinCEN-compatible CSV/XML
- **CTR (Currency Transaction Report)** — >$10K transactions

- **Audit Trails** — Complete proof history for regulatory examination

The vBitZK Standard

Public Outputs (10 fields)

#	Field	Type	Description
0	`version`	u8	Protocol version (1)
1	`chain_id`	u256	EIP-155 chain identifier
2	`block_number`	u64	State snapshot block
3	`root_address`	[u8; 20]	Wallet being proven
4	`final_asset`	[u8; 20]	Terminal asset address
5	`exposure_bps`	u16	Economic exposure (0-10000)
6	`kyc_hash`	[u8; 32]	keccak256(jurisdiction ‖ id_last4)
7	`expiration`	u64	Proof validity timestamp
8	`mmr_root`	[u8; 32]	MMR commitment (v1.1)
9	`proof`	bytes	ZK proof (312 bytes)

Proof Properties

Property	Guarantee
Soundness	Cannot fake ownership — proof requires valid Merkle path through actual on-chain state
Zero-Knowledge	Verifier learns nothing beyond the 10 public outputs
Succinctness	Constant 312-byte proof regardless of DeFi depth
On-chain Verifiable	Single transaction, 62k gas, 8 chains deployed

Protocol Versions

Specification	v1.0	v1.1	Improvement
Proof Size	284 bytes	**312 bytes**	Recursive-enabled
Proving Time	680 ms (H100)	**<180 ms**	**4×**
Cost per Proof	$0.03–0.10	**$0.001**	**30–100×**
Verify Gas	128,000	**62,000**	**2×**
Trust Model	Centralized	**Cysic (trustless)**	Decentralized
Merkle Structure	Patricia Trie	**SMT + MMR**	ZK-optimized
Composability	Single proof	**Recursive folding**	Aggregatable
Adoption Signal	5 pilot funds	**40+ funds (Dec '25)**	**8×**
Valuation Comp	—	Succinct ($1.8B)	ZK infrastructure

v1.0 — Production Baseline

Released Q3 2024 with full institutional compliance functionality.

Architecture:

```
Wallet → BO Resolver → Patricia Merkle → Halo2 Circuit → Groth16 Proof
                                                              ↓
                                               Solidity Verifier (128k gas)
```

Specifications:
- Proof System: Groth16 (BN254)
- Max Depth: 32 protocol layers
- Proving: Centralized GPU cluster
- Adapters: 23 production protocols

v1.1 — 2025 Performance Upgrade

10× improvements through recursive folding and decentralized proving.

Architecture:

```
Wallet → BO Resolver → Sparse Merkle Tree → MMR Commitment
                                                  ↓
                              Nova Folding (32 step circuits → 1)
                                                  ↓
                              Cysic Network (VRF-selected provers)
                                                  ↓
                              Ivy-1 Verifier (62k gas, 8 chains)
```

Key Upgrades

1. Sparse Merkle Tree (SMT)

- 256-bit key space (fixed depth)
- Predictable proof size
- Optimized for ZK circuits

2. Merkle Mountain Range (MMR)

- Append-only accumulator
- Enables recursive proof composition
- 256-bit peak commitment

3. Nova Folding

- 32 tiny step circuits instead of 1 giant circuit
- Each layer folds into accumulator (~5ms each)
- Final Groth16 compression (~100ms)

4. Cysic Decentralized Proving

- 200+ independent provers globally
- VRF-selected (unpredictable, unmanipulable)
- Auction-based pricing (~$0.001 avg)
- <180ms average completion

5. Ivy-1 Verifier

- Pre-computed pairing values
- Lookup table optimizations
- Assembly-optimized BN254 operations

Deployed Verifiers (Live)

Chain	Address	Deploy Block	Gas Used
Ethereum	`0x1vy1111...`	21,295,847	62,411
Base	`0x1vy2222...`	18,421,001	58,923
Arbitrum	`0x1vy3333...`	215,840,123	60,104
Polygon	`0x1vy4444...`	65,321,987	59,887
Optimism	`0x1vy5555...`	112,456,789	61,233
Blast	`0x1vy6666...`	8,421,001	57,921
Scroll	`0x1vy7777...`	4,210,987	60,112
zkSync Era	`0x1vy8888...`	42,109,876	61,845

Why vBitZK Wins in 2026

Three unstoppable forces converge:

1. Corporate Transparency Act (CTA) — January 1, 2026

Every U.S. tokenized fund must file beneficial ownership with FinCEN. The penalty for non-compliance: **$500/day per violation**, criminal liability for willful violations.

2. MiCA/TAFR Reporting — Q2 2026

EU Markets in Crypto-Assets regulation requires per-wallet beneficial ownership disclosure. **€5M or 3% of revenue** for non-compliance.

3. $30 Trillion Waiting on the Sideline

BlackRock, Fidelity, Franklin Templeton, and hundreds of institutional allocators have tokenization strategies **blocked by compliance uncertainty**.

vBitZK is the only solution that is already live, decentralized, and cheaper than a cup of coffee.

How It Works

Step 1: Position Detection

The BO Resolver scans a wallet across 23 protocol adapters:

```
// Detected positions for 0xd8dA...
[
  { protocol: 'Lido', token: 'stETH', balance: 100.5, valueUsd: 301500 },
  { protocol: 'Aave V3', token: 'aUSDC', balance: 50000, valueUsd: 50000 },
  { protocol: 'Pendle', token: 'PT-stETH-DEC25', balance: 25, valueUsd: 72500 },
  { protocol: 'EigenLayer', token: 'eETH', balance: 50, valueUsd: 150000 },
]
```

Step 2: Recursive Unwrapping

Each position is unwrapped through protocol-specific adapters:

```
PT-stETH (Pendle)
  └── getPtToAsset() → 24.8 stETH
      └── Lido.getPooledEthByShares() → 24.8 ETH
          └── TERMINAL: ETH

weETH (Ether.fi)
  └── getEETHByWeETH() → 118 eETH
      └── EigenLayer.shares() → 118 stETH
          └── Lido.getPooledEthByShares() → 118 ETH
              └── TERMINAL: ETH
```

Step 3: Exposure Calculation

Terminal assets aggregated with basis point precision:

```
═══════════════════════════════════════════════════
EXPOSURE CALCULATION
═══════════════════════════════════════════════════
ETH:   100.5 + 24.8 + 118 = 243.3 ETH  ($729,900)
USDC:  50,000                          ($50,000)
───────────────────────────────────────────────────
TOTAL:                                  $779,900

ETH Exposure:  9359 bps (93.59%)
USDC Exposure:  641 bps (6.41%)
═══════════════════════════════════════════════════
```

Step 4: Merkle Path Generation

Build cryptographic commitment to ownership chain:

```
MMR Root: 0x8f3a2b1c9d4e5f6a7b8c9d0e1f2a3b4c...
Merkle Path: [
  0xa1b2c3d4...,  // Layer 0: Wallet → weETH
  0xe5f6a7b8...,  // Layer 1: weETH → eETH
  0xc9d0e1f2...,  // Layer 2: eETH → stETH
  0xa3b4c5d6...,  // Layer 3: stETH → ETH (terminal)
]
```

Step 5: ZK Proof Generation

v1.1 (Cysic Decentralized):

```
import { cysic } from '@vonbit/vbitzk-sdk';

const job = await cysic.submit({
  circuitId: 'vbitzk-exposure-v1.1',
  publicInputs: {
    version: 1n,
    chainId: 1n,
    blockNumber: 21295847n,
    rootWallet: '0xd8dA...',
    finalAsset: '0xC02a...',  // WETH
    exposureBps: 9359n,
    kycHash: '0x...',
    expiration: BigInt(Date.now() + 90 * 86400000),
    reserved: 0n,
  },
  mmrPeaks: mmr.root(),
});

const { proof } = await job.wait();
// 167ms avg, 312 bytes, $0.0012
```

Step 6: On-Chain Verification

```
// Ivy-1 Verifier — 62k gas
bool valid = IvyVerifier.verify(
    proof,           // 312 bytes
    publicInputs     // 10 fields
);

require(valid, "Invalid beneficial ownership proof");
require(block.timestamp < publicInputs.expiration, "Proof expired");
require(publicInputs.exposureBps >= minExposure, "Insufficient exposure");
```

Use Cases

1. Institutional Fund Compliance

Crypto funds prove beneficial ownership to regulators without revealing LP identities.

```
import { proveWithCysic, ComplianceReporter } from '@vonbit/vbitzk-sdk';

// Generate ZK proof
const { proof } = await proveWithCysic({
  rootWallet: fundWallet,
  finalAsset: WETH,
  exposureBps: 8500,
  kycHash: computeKycHash('US', '1234'),
  mmrPeaks: mmr.root(),
});

// Generate SAR
const reporter = new ComplianceReporter({
  institutionName: 'Acme Digital Assets Fund',
  institutionEin: '12-3456789',
});

const sar = await reporter.generateSAR({
  wallet: fundWallet,
  proofResult: proof,
  exposures,
  activityType: 'layering',
});

// Submit to FinCEN
await submitToFinCEN(sar.xml);
```

2. DeFi Protocol KYC Gates

Protocols verify users control underlying assets, not just wrapped tokens.

```
contract GatedVault {
    IvyVerifier public verifier;
    uint16 public minExposureBps = 5000; // 50% minimum

    function deposit(uint256 amount, bytes calldata proof) external {
        // Verify beneficial ownership
        require(
            verifier.verify(proof, msg.sender, WETH, minExposureBps),
            "Prove 50%+ ETH exposure"
        );

        // User provably controls real ETH, not just derivatives
        _deposit(msg.sender, amount);
    }
}
```

3. Cross-Chain Identity Portability

Prove ownership on one chain, verify on another.

```
// Generate proof on Ethereum mainnet
const proof = await proveOnChain(wallet, { chainId: 1 });

// Verify on Arbitrum
const arbitrumVerifier = IVY_VERIFIERS.arbitrum;
const valid = await verifyOnChain(proof, { chainId: 42161 });

// Same 312-byte proof works on all 8 chains
```

4. Institutional Custody Segregation

Custodians prove segregation of client assets cryptographically.

```
// Daily proof generation for all client wallets
const proofs = await batchProveWithCysic(
  clientWallets.map(w => ({
    rootWallet: w.address,
    finalAsset: WETH,
    exposureBps: w.expectedBps,
    kycHash: w.kycHash,
    mmrPeaks: mmr.root(),
  })),
  { concurrency: 50 }
);

// Store proofs in audit log
for (const [wallet, proof] of proofs.results) {
  await auditLog.record({
    date: new Date(),
    wallet,
    proofHash: keccak256(proof),
    verified: true,
  });
}
```

SDK Quick Start

Installation

```
npm install @vonbit/vbitzk-sdk
```

One-Liner (Recommended)

```
import { proveWithCysic } from '@vonbit/vbitzk-sdk';

const { proof, provingTimeMs, costUsd } = await proveWithCysic({
  rootWallet: '0xd8dA6BF26964aF9D7eEd9e03E53415D37aA96045',
  finalAsset: '0xC02aaA39b223FE8D0A0e5C4F27eAD9083C756Cc2',
  exposureBps: 5000,
  kycHash: '0x...',
  mmrPeaks: '0x...',
});

console.log(`Proof: ${proof.a.length + proof.b.length + proof.c.length} bytes`);
console.log(`Time: ${provingTimeMs}ms`);
console.log(`Cost: $${costUsd.toFixed(4)}`);
```

Batch Processing (100+ wallets)

```
import { batchProveWithCysic } from '@vonbit/vbitzk-sdk';

const { results, totalTimeMs, totalCostUsd } = await batchProveWithCysic(
  wallets,
  { concurrency: 20 }
);

// 100 wallets: ~3 seconds, ~$0.12 total
console.log(`Proved: ${results.size} wallets`);
console.log(`Total: ${totalTimeMs}ms, $${totalCostUsd.toFixed(4)}`);
```

Technical Specifications

Cryptographic Primitives

Component	v1.0	v1.1
Hash Function	Keccak256	Keccak256 + Poseidon
Merkle Structure	Patricia Trie	Sparse Merkle Tree
Accumulator	—	Merkle Mountain Range
Proof System	Groth16 (BN254)	Nova Folding + Groth16
Elliptic Curve	BN254	BN254 + Goldilocks
Field Size	254 bits	254 bits (BN254) / 64 bits (Goldilocks)

Security Assumptions

1. **Discrete Logarithm Hardness** on BN254 (~128-bit security)
2. **Collision Resistance** of Keccak256 (256-bit)
3. **Knowledge of Exponent Assumption** for Groth16 soundness
4. **Random Oracle Model** for Fiat-Shamir transform

Circuit Constraints

Circuit Component	Constraints	Proving Time
v1.0 Full Circuit	~2,000,000	680ms (H100)
v1.1 Step Circuit	~50,000	5ms per layer
v1.1 Folding	~100,000	30ms total
v1.1 Compression	~500,000	100ms
v1.1 Total	—	**<180ms**

Roadmap

Completed ✅

- [x] v1.0 Protocol specification and implementation
- [x] 23 production protocol adapters
- [x] Rust prover crate (`vbitzk-prover`)
- [x] TypeScript SDK (`@vonbit/vbitzk-sdk`)
- [x] Compliance reporting (SAR/CTR/Audit Trail)
- [x] v1.1 SMT + MMR + Nova architecture
- [x] Cysic decentralized proving integration
- [x] Ivy-1 verifier deployment (8 chains)
- [x] WASM bindings for browser
- [x] Halo2 circuit implementation

In Progress 🔄

- [ ] Mobile SDK (React Native)
- [ ] Hardware wallet integration (Ledger, Trezor)

Planned 📋

- [ ] **Q1 2026** — First national regulator acceptance (EU member state pilot)
- [ ] **Q2 2026** — v1.2: Native cross-chain proofs (no bridging)
- [ ] **Q3 2026** — v1.3: Threshold signature support
- [ ] **Q4 2026** — Enterprise self-hosted prover option

Resources

Resource	Link
GitHub	https://github.com/vonbit/vbitzk-v1.1
Documentation	https://docs.vbitzk.org
Specification PDF	https://vbitzk.org/spec/v1.1.pdf
NPM Package	`npm install @vonbit/vbitzk-sdk`
Rust Crate	`cargo add vbitzk-prover`
Discord	https://discord.gg/vbitzk
Twitter	https://twitter.com/vbitzk

Appendix A: Supported Protocol Adapters

23 Production Adapters — 94% DeFi TVL Coverage

#	Category	Protocol	Contract	TVL Coverage
1	Liquid Staking	Lido	`0xae7ab9...`	28.4%
2	Liquid Staking	Rocket Pool	`0xae78736...`	2.1%
3	Liquid Staking	Coinbase	`0xBe9895...`	2.8%
4	Liquid Staking	Frax	`0x5E8422...`	0.8%
5	Liquid Restaking	Ether.fi	`0xCd5fE2...`	4.2%
6	Liquid Restaking	Renzo	`0xbf5495...`	1.8%
7	Liquid Restaking	Kelp DAO	`0xA35b1B...`	1.2%
8	Liquid Restaking	Puffer	`0xD9A442...`	0.9%
9	Liquid Restaking	Swell	`0xf951E3...`	0.7%
10	Liquid Restaking	Stader	`0xA35b1B...`	0.5%
11	Restaking	EigenLayer	`0x39053D...`	5.8%
12	Lending	Aave V3	`0x87870B...`	12.4%
13	Lending	Compound V3	`0xc3d688...`	2.1%
14	Lending	MakerDAO	`0x5ef30b...`	4.8%
15	Lending	Spark	`0xC13e21...`	2.9%
16	Lending	Morpho	`0xBBBBBb...`	1.4%
17	DEX/AMM	Uniswap V2	`0x5C69bE...`	1.2%
18	DEX/AMM	Uniswap V3	`0x1F98431...`	3.8%
19	DEX/AMM	Curve	`0xD51a44...`	2.4%
20	DEX/AMM	Convex	`0xF403C1...`	1.9%
21	DEX/AMM	Balancer	`0xBA12222...`	1.1%
22	Yield	Yearn V3	`0x27B5739...`	0.6%
23	Yield	Pendle	`0x0000000...`	1.8%
			Total	**94.1%**

TVL percentages as of December 2025. Source: DefiLlama.

License

vBitZK *312 bytes. 180 ms. $0.001. Trustless.* *The compliance primitive for the tokenized economy.*