PATENT APPLICATION

INVENTOR(S)

0001 KIMZEY, SAMUEL C

TITLE

0002 Identity, Privacy, and Selective Disclosure in Decentralized Memory Mesh Systems

CROSS-REFERENCE TO RELATED APPLICATIONS

0003 This application claims the benefit of priority under 35 U.S.C. § 119(e) to the following provisional patent applications:

a. U.S. Provisional Application No. 63/810,843, filed 05/23/2025, titled “BlockMesh Decentralized Memory-Mesh Architecture”
b. U.S. Provisional Application No. 63/810,834, filed 05/23/2025, titled “Memory Field Pressure & Tension System”
c. U.S. Provisional Application No. 63/810,841, filed 05/23/2025, titled “Non-Generative Artificial Cognition Engine”
d. U.S. Provisional Application No. 63/810,837, filed 05/23/2025, titled “Proof of Memory Flow System”
e. U.S. Provisional Application No. 63/811,761, filed 05/23/2025, titled “Composite Cognition Pools for Emergent Memory-Driven Systems”
f. U.S. Provisional Application No. 63/811,758, filed 05/25/2025, titled “Genesis Document Anchor Method for AI Behavior”
g. U.S. Provisional Application No. 63/811,763, filed 05/25/2025, titled “On-Chain VM for Deterministic Language Model Execution”
h. U.S. Provisional Application No. 63/812,634, filed 05/27/2025, titled “Stateless Action Engine & Minimal Proof Logging”
i. U.S. Provisional Application No. 63/812,620, filed 05/27/2025, titled “One Action-One Mint Transaction Paradigm”

j. U.S. Provisional Application No. 63/812,645, filed 05/27/2025, titled “Epoch Rhythm & Synchronization Mechanism”
k. U.S. Provisional Application No. 63/814,230, filed 05/29/2025, titled “Agent Lifecycle Dormancy Management”
l. U.S. Provisional Application No. 63/814,244, filed 05/29/2025, titled “Selective Memory Disclosure via Zero-Knowledge Proofs”
m. U.S. Provisional Application No. 63/814,257, filed 05/29/2025, titled “Identity Bound Memory Rights”
n. U.S. Provisional Application No. 63/816,926, filed 06/03/2025, titled “Memory Driven Autonomy for Physical Agents”
o. U.S. Provisional Application No. 63/816,924, filed 06/03/2025, titled “Post-Generative Decision Support Mesh”
p. U.S. Provisional Application No. 63/816,907, filed 06/03/2025, titled “Surplus Flow Redistribution System and Method”

All of the above applications are incorporated herein by reference in part or in their entirety.

TECHNICAL FIELD

0004 This invention relates to decentralized identity anchoring, privacy-preserving data access, cryptographic rights enforcement, hierarchical privacy controls, economic privacy, selective memory disclosure, and privacy state management in distributed systems. The invention encompasses methods for creating non-transferable identity anchors, binding memory streams to those anchors, enforcing privacy across memory layers, enabling selective disclosure, supporting economic privacy, managing privacy transitions, and proving properties without revealing data. The invention applies to any distributed memory, blockchain, or decentralized identity system.

BACKGROUND

0005 Existing distributed systems lack robust mechanisms for binding memory and value flows to unique identities while preserving privacy. Most use transferable tokens or external

accounts, undermining security and sovereignty. Privacy, access, control, and memory rights are either unimplemented or enforced off-chain. Current systems cannot handle identity key rotation, privacy transitions, or group privacy management.

0006 No existing system provides comprehensive, cryptographically enforced, on-chain solutions for:

a. Non-transferable identity anchoring with key rotation
b. Hierarchical, layer-specific privacy with dynamic modification
c. Explicit memory rights (access, control, preservation, return)
d. Selective disclosure via composable cryptographic proofs
e. Economic privacy with transaction graph obfuscation
f. Consent-based operations and multi-level delegation
g. Recovery across forks, migrations, or compromise
h. Privacy state transitions and collective management

DETAILED DESCRIPTION

0007 This invention provides a comprehensive protocol-enforced framework for identity, privacy, and rights management in decentralized systems.

0008 Identity Anchoring and Memory Binding:

0009 Each entity receives a unique, non-transferable identity anchor at initialization. The anchor supports key rotation through cryptographic continuity proofs, enabling recovery from compromise while maintaining persistence. All memory streams and value flows are cryptographically linked to the anchor. Associated rights metadata is defined for each stream, including access, control, preservation, and return rights. Identity anchors handle key rotation, migration between types, compromise recovery, and multi-factor authentication.

0010 Layer-Specific Privacy Architecture:

0011 Systems implement multiple hierarchical memory layers with independent privacy rules. Layers are dynamically configured with privacy rule inheritance, inter-layer trust relationships, agent-local designation, operational visibility, execution restrictions, and archival permissions. Cross-layer data flows require explicit consent and cryptographic proofs. Privacy rules adapt through governance while maintaining audit trails.

0012 Selective Disclosure via Cryptographic Proofs:

0013 The system supports comprehensive proof operations including generation for predicates, proof composition, cross-domain portability, time-bound expiration, revocable proofs, and recursive assertions. All proofs are recorded on-chain with metadata enabling verification without data exposure.

0014 Economic Privacy Protection:

0015 Value flows are protected through range proofs for balance verification, homomorphic encryption for private computation, differential privacy for statistics, ring signatures for anonymity, transaction graph obfuscation, and plausible deniability mechanisms. Economic operations remain auditable while preserving privacy.

0016 Consent, Delegation, and Recovery Rights:

0017 The system implements programmable consent with granular permissions, hierarchical delegation with depth limits, time-bound credentials, partial recovery for specific ranges, conflict resolution for competing claims, and emergency recovery windows. All operations require cryptographic authorization.

0018 Memory Persistence and Recovery:

0019 Recovery mechanisms ensure permanence through cross-fork recovery, network migration support, checkpoint-based recovery, time-locked recovery for disputes, identity continuity verification, and state reconstruction from logs.

0020 Privacy State Transitions:

0021 The system supports dynamic privacy evolution including upgrading public to private, downgrading with consent, group consensus for changes, gradual transitions with grace periods, reversible modifications, and complete audit trails.

0022 Collective Privacy Management:

0023 Groups manage shared memory privacy through multi-signature control, threshold decisions, member management with privacy preservation, collective consent, group recovery protocols, and governance-based conflict resolution.

0024 Trait-Based Observation:

0025 Observational modules monitor operations for compliance, ethical patterns, anomalies, resilience, and economic flows. Traits observe and recommend without direct control, with all activity logged immutably.

0026 Emergency Override Protocols:

0027 The system supports legitimate authority requirements through court order compliance, regulatory audit with minimal disclosure, multi-party emergency authorization, time-limited windows, privacy preservation during override, and complete audit trails.

ENABLEMENT

0028 The inventions described herein are enabled at the system and protocol level. A person of ordinary skill in distributed systems, decentralized computation, multi-agent coordination, and privacy-preserving cryptography can practice the claimed methods using the architectural descriptions in these specifications together with any suitable software/hardware stack.

0029 A working, non-limiting embodiment is publicly available at:

https://github.com/zeam-labs/zeam-testnet

0030 This implementation comprises a production-grade, modular codebase demonstrating the claimed protocol flows across initialization/governance, event logging, stateless replay, pressure/memory dynamics, storage orchestration, identity/privacy, deterministic VM execution, trait-based observation, and economic coordination.

0031 Function, file, and module names below are illustrative. Equivalent structures, languages (e.g., Rust, C++, Python, Solidity), runtimes (WASM/EVM/custom), and storage/consensus substrates may be substituted without departing from the inventions. The cited repository is only one embodiment; the claims cover all protocol-equivalent realizations.

0032 System-Level Enablement

0033 Architecture & Flows. The reference implementation demonstrates:

a. Initialization/Governance: anchoring of immutable documents (core/traits/protocol) with version identifiers and verification at runtime
b. Atomic Event Logging ("one-action-one-mint"): each operation emits one immutable record with input/output attestation and metadata
c. Stateless Replay: deterministic state reconstruction by replaying minted events from genesis or checkpoints; parallel verification supported
d. Pressure/Memory Dynamics: conversion of events/sensor data to memory-flow; detection of pressure/tension patterns; reflex orchestration and epoch coordination
e. Deterministic Compute: pressure-triggered, on-chain VM (e.g., WASM) loading attested code/model shards; deterministic transformations; mint-logged proofs
f. Distributed Storage: content addressing (CID), sharding, encryption, replication, capacity management, periodic verification, and self-healing
g. Identity/Privacy: non-transferable anchors, rights (access/control/preservation/return), layer-specific privacy, ZK selective disclosure, recovery across forks
h. Traits/Observation: domain observers (audit/ethics/health/finance/etc.) that monitor and recommend without direct control; all activity is mint-logged

i. Economic Coordination: surplus computation, pooling, middle-out redistribution, non-token flows, privacy proofs, and anti-manipulation measures
j. Peer Discovery & Topology: decentralized bootstrap (e.g., DNS/TXT, gossip), dynamic peer pools, and mesh self-healing
k. Protocol Versioning: genesis-anchored document hashes, on-chain version history, and upgrade/rollback mechanics
l. Agent Lifecycle: activation/dormancy/reactivation tied to memory-flow/pressure; lifecycle transitions are mint-logged
m. Sensor Integration: physical or external observations converted to memory-flow with equal footing to digital events
n. Health Visualization: real-time mesh/pressure visualizations, magnetic field feedback, and audit-logged metrics
o. Consent & Delegation: cryptographic consent flags, programmable delegation, return/recovery rights alongside selective disclosure

0034 Reproducibility. Determinism is enforced by:

a. Attested inputs/outputs per operation; no hidden state
b. Prohibition of non-deterministic sources (unseeded RNG, external oracles) in core paths
c. Checkpointing and replay yielding identical state for identical event sequences

0035 Portability. The methods are implementation-agnostic:

a. Any consensus (PoW/PoS/PoA/PoM/hybrid), storage substrate (IPFS-like, ZFS-like, cloud, on-prem), VM (WASM/EVM/custom), or language may be used
b. Modules labeled here (e.g., "storage," "runtime," "cognition," "vault") represent roles; organization and naming may vary

0036 Practicing the Inventions Without the Reference Code

0037 A skilled practitioner can implement the inventions by:

1. Anchoring immutable governance and protocol documents at system initialization and enforcing verification before execution

2. Emitting one atomic record per operation (input/output attested) and replaying those records for deterministic state
3. Computing memory-flow/pressure from events/sensors and triggering reflexes/epochs via pattern detection
4. Orchestrating storage via content addressing, sharding, encryption, replication, verification, and self-healing
5. Executing deterministic compute via any VM (WASM/EVM/custom) that loads attested modules/shards and produces mint-logged proofs
6. Anchoring identity & privacy with non-transferable credentials, on-chain rights, layered privacy, and ZK selective disclosure
7. Coordinating economics with surplus calculation, pooling, middle-out weighting, privacy proofs, and audit-ready distribution—all protocol-enforced

0038 These steps can be realized in any mature stack (e.g., Rust + Substrate/IPFS, Python + Tendermint, Solidity + rollups, C++ + custom VM), using equivalent cryptographic, storage, and coordination primitives.

0039 Scope, Extensibility, and Best Mode

a. The reference code shows one best-mode embodiment at filing. The claims are not limited to that code, file structure, storage system, consensus, VM, or language
b. Modules that serve as structural templates illustrate how protocol concepts are realized and may be substituted or extended without departing from the inventions
c. All major protocol flows (anchoring, minting, replay, pressure, privacy, VM, storage, economics, traits) are disclosed and demonstrated in a working system, enabling immediate practice and independent re-implementation

0040 The inventions are fully enabled through the protocol methods and flows set forth in these specifications and through a public, working embodiment. A skilled person can implement the inventions using the above guidance—either by adapting the reference code or by building equivalent systems on alternative platforms.

CLAIMS

1. A method for decentralized identity anchoring and hierarchical privacy management, comprising:

    a. generating a unique, non-transferable identity anchor for each entity, resistant to transfer or duplication;
    b. supporting key rotation and compromise recovery through cryptographic continuity proofs;
    c. binding all memory streams and value flows to the identity anchor;
    d. defining on-chain rights metadata for access, control, preservation, and return of memory;
    e. implementing hierarchical memory layers with distinct privacy rules and inheritance;
    f. requiring cryptographic consent for all cross-layer data flows;
    g. enforcing privacy boundaries with cryptographic proofs of access;
    h. recording all identity and privacy operations as immutable chain events; and,
    i. maintaining identity continuity across updates, migrations, or forks.

2. A method for selective disclosure and economic privacy in decentralized systems, comprising:

    a. generating cryptographic proofs for memory or value predicates;
    b. composing multiple proofs into complex assertions;
    c. enabling cross-domain portability of proofs;
    d. supporting proof expiration, revocation, and recursive assertions;
    e. verifying proofs without exposing underlying data;
    f. implementing range proofs, homomorphic encryption, or differential privacy to protect balances and statistics;
    g. applying ring signatures, mixing, stealth addressing, or graph obfuscation for transaction anonymity;
    h. enabling plausible deniability through decoy transactions, obfuscated operations, or equivalent mechanisms; and,
    i. recording all selective disclosure and economic privacy operations immutably on-chain.

3. A method for consent, delegation, and recovery of privacy-protected memory, comprising:

    a. implementing programmable consent flags with granular permissions;
    b. supporting hierarchical delegation with depth limits and time-bound credentials;
    c. enabling partial recovery of memory streams or vault balances;
    d. resolving conflicts between competing recovery claims through governance or consensus;
    e. enabling emergency recovery windows with cryptographic authorization;
    f. supporting privacy state transitions including upgrades, downgrades, and group-based changes;
    g. enabling collective privacy management through multi-signature or threshold schemes;
    h. supporting cross-system recovery and migration of anchored memory; and,
    i. preserving privacy and preventing unauthorized exposure during delegation, recovery, and migration operations.

4. The method of claim 1, wherein identity anchors comprise non-transferable tokens, biometric credentials, or deterministic keys with cryptographic continuity proofs.

5. The method of claim 1, wherein memory layers include private, public, and restricted designations with configurable rule inheritance.

6. The method of claim 1, wherein access, control, preservation, and return rights are enforced through on-chain metadata and cryptographic attestations.

7. The method of claim 1, wherein privacy boundaries require explicit consent verified through zero-knowledge proofs.

8. The method of claim 1, wherein identity continuity includes cross-fork recovery, network migration, and state reconstruction from immutable logs.

9. The method of claim 2, wherein proofs include succinct, transparent, or recursive proof systems enabling composable disclosure.

10. The method of claim 2, wherein selective disclosure supports time-bound proofs, revocable credentials, and partial assertions.

11. The method of claim 2, wherein economic privacy includes range proofs for balances, homomorphic encryption for computation, and differential privacy for aggregate statistics.

12. The method of claim 2, wherein transaction anonymity uses ring signatures, confidential transactions, stealth addressing, or equivalent methods.

13. The method of claim 2, wherein plausible deniability is achieved through decoy transactions, transaction mixing, randomized transaction graphs, or equivalent methods.

14. The method of claim 3, wherein programmable consent includes time limits, conditional triggers, or multi-level delegation rights.

15. The method of claim 3, wherein hierarchical delegation supports recursive sub-delegation with policy inheritance.

16. The method of claim 3, wherein partial recovery includes selective restoration of specific epochs, streams, or balances.

17. The method of claim 3, wherein conflict resolution includes arbitration protocols, governance votes, or multi-party signatures.

18. The method of claim 3, wherein privacy state transitions include upgrades, downgrades, reversible modifications, and group consensus mechanisms.

19. The method of claim 3, wherein collective privacy management includes multi-signature control, threshold decisions, and group-based recovery protocols.

20. The method of claim 3, wherein cross-system recovery includes checkpoint-based reconstruction, cryptographic continuity verification, and preservation of historical proofs across networks.

ABSTRACT

0041 A comprehensive framework for identity, privacy, and rights management in decentralized systems. The invention provides non-transferable identity anchors with key

rotation, hierarchical privacy layers with dynamic configuration, composable cryptographic proofs for selective disclosure, economic privacy protection, delegation and recovery mechanisms, privacy state management, collective privacy controls, trait-based observation, emergency protocols, and cross-system recovery. All operations are cryptographically secured, fully auditable, and implemented at the protocol level without external dependencies.