PATENT APPLICATION

INVENTOR(S)

[0001] KIMZEY, Samuel C

TITLE

[0002] Identity-Bound Memory Rights

TECHNICAL FIELD

[0003] The invention relates to identity management and personal data sovereignty in decentralized systems, specifically to methods for securing memory rights via persistent, non-transferable identity anchors on blockchain networks.

BACKGROUND

[0004] Existing decentralized ledgers record historical transactions, but lack robust mechanisms for irrevocably binding those records to unique identities in a way that enforces control, access, and preservation rights over personal memory. This absence creates risks of unauthorized access, tampering, and disassociation from the rightful owner. A method is needed to ensure memory streams are anchored immutably and enforceably to a sovereign identity.

SUMMARY

[0005] The Identity-Bound Memory Rights method comprises:

1. Generating a non-transferable identity anchor (e.g., soulbound address or token) for each user or entity.

2. Binding memory streams – chronological sequences of on-chain events – to the identity anchor via cryptographic links.

3. Assigning explicit rights over memory streams, including Access Right, Control Right, and Preservation Right.

4. Enforcing rights by validating identity signatures before permitting access or modification.

5. Logging all unauthorized access attempts or rights violations as immutable audit events.

DETAILED DESCRIPTION

[0006] Each user or entity is issued a persistent, non-transferable identity anchor – such as a soulbound token or deterministic address – during genesis or account instantiation. On-chain memory events minted by or about that identity are cryptographically linked to the anchor. Associated rights metadata defines who may read, write, or delegate portions of the stream. All interactions require cryptographic signatures validated against the anchor and its rights. Violations—such as signature mismatches or overreach—are automatically logged as immutable audit events, ensuring traceability and integrity.

METHOD FLOW

1. Identity Anchor Creation – Generate a non-transferable identity object bound to the user or entity.

2. Memory Event Linking – For each new memory event, cryptographically associate it with the identity anchor.

3. Rights Assignment – Define and store on-chain rights metadata (Access, Control, Preservation).

4. Rights Enforcement – Verify all access/modification requests against the anchor and rights metadata.

5. Audit Logging – Mint immutable audit entries upon any failed or unauthorized interaction.

NARRATIVE WORKED EXAMPLE

[0007] Alice creates an identity anchor 'ID_Alice'. She performs two memory actions: posting a health record and a personal note. Each event is minted on-chain with 'ID_Alice' in the metadata. Later, Alice requests her personal history, signs the request with her private key, and retrieves the stream. An unauthorized request by Eve fails signature verification and generates an audit event 'UnauthorizedAccessAttempt(ID_Alice, Eve)'.

ALGORITHMIC WORKED EXAMPLE

[0008] Pseudocode:

```
// Step 1: Create a non-transferable identity anchor
idAnchor := CreateIdentityAnchor(userPublicKey)
// Step 2: Mint memory events bound to the anchor
for _, event := range memoryEvents {
    event.Metadata["identity"] = idAnchor
    event.Hash = sha256(event.Data + idAnchor)
```

```
    MintEvent(event)
}
// Step 3: Handle an incoming memory access request
request := ReceiveRequest()
// Step 4: Verify rights before granting access
if VerifySignature(request.Signature, idAnchor) &&
   HasRight(idAnchor, "Access") {
    return FetchMemoryStream(idAnchor)
}
// Step 5: If verification fails, mint an immutable audit log
MintAuditEvent(
    type = "AccessViolation",
    identity = idAnchor,
    origin = request.Origin,
    timestamp = NowUTC()
)
```

POTENTIAL EMBODIMENTS

1. Biometric or multi-factor verification bound to the identity anchor.

2. Use of zero-knowledge proofs to permit selective access without identity disclosure.

3. Multi-chain anchors allowing distributed memory across chains with shared verification.

4. Revocable delegation of memory rights via temporary credentials.

IMPLEMENTATION NOTES

[0009] All identity anchors and memory events conform to a one-action-one-mint paradigm. Rights metadata and audit logs are stored on-chain; enforcement is handled entirely within protocol modules, without off-chain dependencies. Trait logic may observe but cannot override these rights.

CLAIMS

1. A method for binding memory streams to persistent identities on a blockchain, comprising:
    a. generating, by a processor, a non-transferable identity anchor for a user or entity;
    b. linking, by the processor, each memory event to the identity anchor via cryptographic metadata;
    c. defining, by the processor, rights metadata (Access, Control, Preservation) associated with the identity anchor;
    d. verifying, by the processor, a request signature against the identity anchor and rights metadata before memory access;
    e. minting, by the processor, an immutable audit event upon rights violation.

2. The method of claim 1, wherein the identity anchor is a soulbound token minted at account creation.

3. The method of claim 1, wherein rights metadata is stored as on-chain attributes linked to the identity anchor.

4. The method of claim 1, further comprising allowing delegation of rights via revocable credential events.

5. The method of claim 1, wherein all verification is performed within on-chain protocol modules without external dependencies.

6. The method of claim 1, wherein rights enforcement occurs without reliance on persistent module state, using only protocol-defined input validation at the time of memory interaction.

7. The method of claim 1, further comprising one or more observational traits that interpret access patterns and log ethical or structural violations without modifying or blocking access directly.

8. The method of claim 1, wherein rights violations trigger the automatic minting of immutable audit events without human intervention or discretionary oversight.

9. The method of claim 1, further comprising the issuance of signed, revocable credentials that temporarily delegate specific memory rights to authorized third parties.

ABSTRACT

[00010] A system and method for binding immutable memory streams to persistent, non-transferable identities on a blockchain. Each identity anchor governs a memory log composed of one-action-one-mint events, with explicit rights for access, control, and preservation. Unauthorized access attempts are rejected and immutably logged, ensuring that personal memory remains sovereign, tamper-proof, and enforceably linked to its origin.