Chain of Custody Delegation in Multi-Agent Systems

Ramprasad Anandam Gaddam
Vouch Protocol Project
https://vouch-protocol.com
ram@vouch-protocol.com

January 2026

Abstract

This paper addresses the problem of maintaining cryptographic accountability when AI agents delegate tasks to other agents. We present a delegation chain protocol that enables verifiable tracking of action lineage through linked cryptographic attestations. Each agent in a delegation chain signs their portion of work while referencing their delegator's attestation, creating an auditable and tamper-evident trail of responsibility. We formalize the delegation model, define the chain structure, and analyze the security properties achieved. The protocol enables enterprise compliance, liability attribution, and transparent multi-agent coordination.

This paper is part of the Vouch Protocol Defensive Disclosure Series. Full index: https://vouch-protocol.com/docs/disclosures/

Keywords: delegation, chain of custody, multi-agent systems, cryptographic attestation, accountability

1 Introduction

Modern AI systems increasingly operate as multi-agent architectures where specialized agents collaborate to accomplish complex tasks. A personal assistant agent may delegate research to a specialized research agent, which in turn may delegate web scraping to a data collection agent. This delegation pattern creates a fundamental accountability challenge: when an action causes harm or exceeds authorized scope, determining which agent is responsible becomes difficult without a verifiable audit trail.

1.1 Problem Statement

Consider a multi-agent system where Agent A delegates to Agent B, which further delegates to Agent C. If Agent C performs an unauthorized action, the following questions arise:

1. Did Agent B authorize Agent C to perform this action?
2. Did Agent A authorize Agent B with sufficient scope?
3. Can any party deny their role in the delegation chain?

1.2 Contributions

This paper makes the following contributions:

- Formal model for delegation in multi-agent systems

- Protocol for cryptographically linked delegation chains
- Scope constraint propagation mechanism
- Security analysis of chain integrity properties

2 Background

2.1 Delegation in Distributed Systems

Definition 1 (Delegation). *A delegation is a tuple* $D = (\mathsf{delegator}, \mathsf{delegatee}, \mathsf{scope}, \mathsf{constraints}, \sigma)$ *where the delegator authorizes the delegatee to perform actions within the specified scope, subject to constraints, signed by* σ*.*

Delegation differs from simple authorization in that the delegatee acts *on behalf of* the delegator, creating a chain of responsibility.

2.2 Hash Chains

Definition 2 (Hash Chain). *A hash chain is a sequence of records where each record* r_i *includes the cryptographic hash of the previous record:*

$$r_i.\mathsf{prev\_hash} = H(r_{i-1}) \tag{1}$$

where H *is a collision-resistant hash function.*

Hash chains provide tamper-evidence: modifying any record invalidates all subsequent hashes [Merkle, 1988].

3 Delegation Chain Protocol

3.1 Chain Structure

Definition 3 (Delegation Chain). *A delegation chain* $C = (D_0, D_1, \ldots, D_n)$ *is a sequence of delegations where:*

$$D_i.\mathsf{parent\_hash} = H(D_{i-1}) \quad \forall i > 0 \tag{2}$$

and each D_i *is signed by* $D_{i-1}.\mathsf{delegatee}$*.*

3.2 Delegation Token Structure

```
{
  "type": "delegation",
  "delegator": "did:web:agent-a.example.com",
  "delegatee": "did:web:agent-b.example.com",
  "scope": {
    "action": "research",
    "max_queries": 10,
    "allowed_domains": ["*.wikipedia.org"]
  },
  "parent_hash": "sha256:abc123...",
  "created_at": "2026-01-10T12:00:00Z",
  "expires": "2026-01-10T12:05:00Z"
}
```

3.3 Scope Propagation

Property 1 (Scope Monotonicity). *A delegatee cannot grant more authority than they possess:*

$$\mathsf{scope}(D_i) \subseteq \mathsf{scope}(D_{i-1}) \tag{3}$$

This property ensures that delegation chains cannot escalate privileges.

3.4 Chain Verification

To verify action a performed by Agent C at chain depth n:

1. Obtain Agent C's attestation A_n for action a
2. Verify $\mathsf{signature}(A_n)$ using C's public key
3. Follow $A_n.\mathsf{parent\_hash}$ to obtain D_{n-1}
4. Verify $a \in \mathsf{scope}(D_{n-1})$
5. Recursively verify delegation chain to root
6. Verify root delegator has sufficient authority

4 Security Analysis

Property 2 (Chain Integrity). *Any modification to a delegation in the chain invalidates all subsequent parent hashes.*

Property 3 (Non-Repudiation). *Each delegation is signed by the delegator. The signature serves as cryptographic evidence of authorization.*

Property 4 (Scope Enforcement). *Actions outside delegated scope can be detected by comparing action parameters against scope constraints.*

Property 5 (Temporal Validity). *Expired delegations are rejected, limiting the window for misuse.*

5 Related Work

SPIFFE/SPIRE provides workload identity but lacks delegation chaining with scope constraints.

OAuth 2.0 Token Exchange [Hardt, 2012] supports token delegation but without cryptographic chain linking.

Verifiable Credentials [Sporny et al., 2022] provide credential chaining; our work extends this to action delegation.

6 Implementation

Reference implementation available at `https://github.com/vouch-protocol/vouch`.

The implementation provides:

- Delegation token creation and signing
- Chain verification with scope checking
- Integration with PAD-001 identity tokens

7 Conclusion

This paper presented a cryptographic delegation chain protocol for multi-agent systems. By linking attestations through hash chains and enforcing scope monotonicity, the protocol enables verifiable accountability across delegation hierarchies. Future work includes dynamic scope renegotiation and revocation mechanisms.

Cryptographically Signed Document

Signed by: Ramprasad Anandam Gaddam (github:rampyg)
Verify: https://v.vouch-protocol.com/p/pad002

This document's authenticity can be verified by computing its SHA-256 hash and checking against the signature registered at the verification URL above.

References

D. Hardt. The OAuth 2.0 Authorization Framework. RFC 6749, IETF, October 2012. URL https://www.rfc-editor.org/rfc/rfc6749.

Ralph C. Merkle. A Digital Signature Based on a Conventional Encryption Function. In *Advances in Cryptology — CRYPTO '87*, pages 369–378. Springer, 1988.

Manu Sporny, Grant Noble, Dave Longley, Daniel Burnett, Brent Zundel, and Kyle Den Hartog. Verifiable Credentials Data Model v1.1. Recommendation, W3C, March 2022. URL https://www.w3.org/TR/vc-data-model/.