March 13, 2026

To: AI-Identity@nist.gov

Re: Public Comment — Accelerating the Adoption of Software and AI Agent Identity and Authorization

---

Public Comment on NIST NCCoE Concept Paper

Accelerating the Adoption of Software and AI Agent Identity and Authorization

Submitted by: Hernán Alfredo Capucci

Author, Agent Manifest Specification

ORCID: 0009-0008-7216-3032

agent-manifest-spec.org

DOI: 10.5281/zenodo.18833956

---

1. Introduction

This comment is submitted in response to NIST NCCoE's request for public input on the concept paper "Accelerating the Adoption of Software and AI Agent Identity and Authorization" (February 2026). The NCCoE concept paper correctly highlights the growing need for identity and authorization mechanisms for AI agents operating in enterprise environments.

I am the author of Agent Manifest, an open declarative specification that addresses precisely the problem NIST has identified: autonomous AI agents currently operate without a standardized mechanism to declare their identity, operational boundaries, and chain of human accountability before interacting with external systems.

Agent Manifest was developed as a minimal, open standard grounded in a foundational principle:

> *Autonomous systems should declare their boundaries before action.*

This comment presents Agent Manifest as a concrete, available implementation that may inform NIST's demonstration project — particularly in the areas of agent identification, transparency, and pre-execution declaration.

2. The Declaration Gap NIST Has Correctly Identified

NIST's concept paper accurately identifies that existing identity frameworks — OAuth, OpenID Connect, SPIFFE/SPIRE — were designed for human users and static software services, not for autonomous agents with dynamic, potentially irreversible decision-making capabilities.

There is a specific gap these frameworks do not address: the moment before an agent acts. Current protocols focus on authentication ("is this agent who it claims to be?") and authorization ("is this agent permitted to perform this action?"). But they do not address declaration: the structured, public statement of what an agent is, what it may and may not do, and who is responsible for it.

This distinction matters because:

- Authentication verifies identity at runtime. Declaration makes identity inspectable before runtime.

- Authorization enforces permissions. Declaration communicates intended boundaries to all parties — including other agents, auditors, and users.
- Governance frameworks operate after the fact. Declaration operates before execution begins.

Agent Manifest was designed specifically to fill this pre-execution declaration layer.

3. Agent Manifest: A Concrete Implementation

3.1 What it is

Agent Manifest is an open declarative specification that defines a minimal JSON structure through which an AI agent declares:

- Identity: agent_id, agent_name, agent_version, owner
- Purpose: primary operational domain and human-readable description
- Boundaries: explicitly declared forbidden_actions
- Autonomy level: a structured scale from supervised to fully autonomous
- Risk profile: declared risk surface (low, medium, high, critical)
- Data handling commitments: whether personal data is stored or processed
- Stopping authority: who can halt the agent and through what mechanism
- Audit surface: logging level and reconstructability of actions
- Contact: responsible human or organization

The specification is intentionally minimal and execution-agnostic. It defines what an agent declares, not what it does.

3.2 Key design principles

1. Declaration Layer Only: Agent Manifest does not execute agents, enforce compliance, or certify safety. It standardizes the declaration surface.
2. Minimal by design: The schema requires only the fields necessary for transparent identification. Complexity belongs in enforcement layers built on top.
3. Open and composable: The specification is published under CC BY 4.0 and is designed to complement — not replace — existing identity standards such as OAuth and OpenID Connect.
4. Human chain of responsibility: Every manifest requires declaration of a stopping authority and a responsible owner, preserving human accountability in the chain.

3.3 Ecosystem infrastructure

Agent Manifest is not only a specification. A working ecosystem has been built around it:

- Canonical specification (v1.0): agent-manifest-spec.org — includes normative JSON Schema, design rationale, core principles, and stability policy.
- Ambassador (generator): A five-step conversational tool that allows any developer to produce a valid manifest in under five minutes. Available at: hernancapucci.github.io/agent-manifest-ambassador/
- Registry: A public discovery endpoint at /.well-known/agent-manifest-registry.json, following the pattern established by robots.txt and well-known URI conventions.

- Dataset: A public repository of registered manifests structured as manifests/YYYY/MM/agent-name.json, constituting an auditable public record.
- Boundary Handshake: An experimental protocol layer for agent-to-agent declaration exchange before interaction.
- Philosophical foundation: The \u2208 Principle (The Destination of Actions), published with DOI at Zenodo, which grounds the specification in a theory of public authorship and responsible autonomy.

3.4 Academic record

The specification and its companion analysis are archived on Zenodo with permanent DOIs:

- Core Declarative Specification: https://doi.org/10.5281/zenodo.18833956
- Companion analysis: https://doi.org/10.5281/zenodo.18834845
- The \u2208 Principle (EN): https://doi.org/10.5281/zenodo.18945860

These records provide a stable, citable reference for any standards body or research project that wishes to reference Agent Manifest.

4. Responses to NIST's Specific Questions

4.1 What metadata is essential for an AI agent's identity?

Based on the Agent Manifest specification, the minimum viable identity surface for an AI agent includes:

- A unique agent identifier (agent_id) and human-readable name
- Version information for lifecycle tracking
- A declared owner — individual or organization — who assumes public accountability
- A primary operational purpose code and description
- An autonomy level indicator (the degree to which the agent acts without human approval)
- A stopping authority declaration — who can halt the agent and how

This minimum set enables identification, accountability linkage, and human oversight — the three pillars NIST has identified as foundational.

4.2 How should agents be recognized in enterprise architectures?

Agent Manifest proposes a complementary approach to OAuth-based authentication: a publicly accessible declaration file (manifest.json) that any system — human or machine — can read before initiating interaction.

This follows the well-known URI pattern already established in web infrastructure (robots.txt, /.well-known/). An enterprise architecture could require agents to publish a manifest at a discoverable endpoint as a precondition for interaction — independent of, and complementary to, runtime authentication flows.

4.3 What standards or protocols should NIST consider?

NIST's concept paper lists MCP, OAuth 2.0, OpenID Connect, and SPIFFE/SPIRE as protocols under consideration. Agent Manifest proposes a complementary, upstream layer: the pre-execution declaration standard.

These layers are not competing. A complete agent identity architecture might look like:

5. Declaration Layer (Agent Manifest): What does the agent declare about itself before acting?
6. Authentication Layer (OAuth, OpenID Connect, SPIFFE): Is the agent who it claims to be at runtime?
7. Authorization Layer (policy engines, NGAC): Is the agent permitted to perform this specific action?
8. Audit Layer (logging, SIEM): What did the agent actually do, and can it be reconstructed?

Agent Manifest occupies Layer 1 — currently absent from NIST's stack — and is designed to feed into Layers 2, 3, and 4.

4.4 What use cases exist for AI agent identity?

Agent Manifest was designed with three primary use cases:

- Developer accountability: Any individual or organization building an agent can publish a manifest to declare responsibility before deployment.
- Agent-to-agent trust negotiation: Two agents can exchange manifests before interaction to assess whether their declared boundaries are compatible — this is the Boundary Handshake protocol.
- Audit and governance: Regulators, enterprises, and users can inspect a manifest to understand an agent's declared behavior surface without requiring access to its internal implementation.

5. Recommendation to NIST

I respectfully recommend that NIST consider evaluating Agent Manifest as a reference implementation for the declaration layer of AI agent identity.

Specifically:

9. Include a pre-execution declaration standard in the demonstration scope. The current concept paper focuses on authentication and authorization but does not address what agents declare before they act. This layer is foundational to transparency and accountability.
10. Consider the /.well-known/agent-manifest pattern as a low-friction discovery mechanism analogous to robots.txt — requiring no new infrastructure for enterprises already operating on web standards.
11. Evaluate the autonomy level field as a structured mechanism for human-in-the-loop policy: agents with autonomy level 1 (supervised) could trigger different authorization policies than agents at level 4 (fully autonomous).
12. Treat declaration and authentication as separate concerns. Declaration is a public commitment. Authentication is a runtime verification. Conflating them creates unnecessary friction for lower-risk agent deployments.

6. Conclusion

The problem NIST has identified is real and urgent: AI agents are becoming operational actors in enterprise environments without a standardized mechanism for declaring who they are, what they may do, and who is responsible for them.

Agent Manifest is a working, documented, openly licensed response to this problem. It is minimal by design, composable with existing standards, and grounded in a clear philosophical principle: declaration before interaction.

I welcome the opportunity to contribute further to NIST's demonstration project and am available to provide technical documentation, answer questions, or participate in listening sessions.

Contact and References

Hernán Alfredo Capucci

Author, Agent Manifest Specification

ORCID: 0009-0008-7216-3032

Primary references:

Specification: agent-manifest-spec.org

DOI (spec): 10.5281/zenodo.18833956

DOI (companion): 10.5281/zenodo.18834845

∈ Principle: hernancapucci.github.io/e-principle/

GitHub: github.com/hernanCapucci/agent-manifest

Boundary Handshake: github.com/hernanCapucci/boundary-handshake

---