Subject: Public Comment on NCCoE Concept Paper: Execution-Boundary Governance for AI Agents
Date: Thursday, March 26, 2026 at 9:41:36 PM Central Daylight Time
From: Vincent Nijjar <vincent@zlar.ai>
To: AI-Identity@nist.gov <AI-Identity@nist.gov>
Attachments: Appendix_ZLAR_Map.docx, ZLAR_Glossary.docx

Dear NCCoE Team,

My name is Vincent Nijjar. My professional background is in economics, finance, and asset management, serving wealth advisory organizations across Canada. I approach this topic as a systems thinker focused on how rapidly advancing AI capabilities are changing operational, governance, and accountability requirements in real time. I am responding because I believe AI can create real abundance and productivity, but only if human dignity, human judgment, and human accountability remain central to how these systems are designed and governed.

I am writing because I believe your concept paper identifies the right problem, but one architectural gap deserves even sharper emphasis: the missing layer in agentic systems is execution-boundary governance.

Current standards and emerging protocols can help establish who an agent is, how it authenticates, and what delegated authority it carries. But they do not, by themselves, answer the most important operational question: should this exact action be allowed right now, under this context, under this delegated authority, with evidence that can later be verified?

Authenticated is not authorized. Delegated is not governed. Reasoning is not permission.

That is the problem I have been working on through ZLAR, an execution-boundary governance layer for autonomous AI agents. ZLAR is built around a simple design principle: the governance gate should have no intelligence. The gate should not interpret natural language, should not reason about whether the model “meant well,” and should not become another model-layer attack surface. It should evaluate structured action requests against explicit policy and either allow, deny, or escalate them.

That property matters because prompt injection, context poisoning, and authority confusion attack the reasoning layer. A deterministic external gate does not solve those reasoning failures directly. What it does do is keep compromise of reasoning from automatically becoming compromise of action.

In the current ZLAR implementation, every agent action is intercepted at the execution boundary, checked against signed policy, and recorded in a hash-chained audit trail with cryptographic integrity. Each audit entry carries a FIPS 186-5 EdDSA digital signature satisfying SP 800-53 AU-10 (Non-repudiation), and entries are linked via FIPS 180-4 SHA-256 hash chaining satisfying AU-9(3) (Cryptographic integrity protection). The repository currently implements two enforcement surfaces that share the same policy and evidence model: a Bash gate that hooks into agent tool use, and an MCP gate that sits between MCP clients and MCP servers. The governing idea is the same in both cases: the agent does not volunteer to be governed; it is governed by architecture.

I believe this maps especially well to the concept paper's questions on authorization, access delegation, logging and transparency, and prompt-injection impact reduction.

For NCCoE, I would strongly encourage a demonstration architecture that includes the following:

1. Per-action authorization at the moment of execution, not only at login, startup, or token issuance.
2. A non-bypassable enforcement point outside the agent process.
3. Bounded human-to-agent delegation with explicit scope, time, and sensitivity constraints.
4. Verifiable governance receipts binding together agent identity, delegating identity, requested action, policy version, decision, and timestamp.
5. Tamper-evident auditability suitable for enterprise, regulated, and public-sector review.
6. Prompt-injection containment through external policy enforcement, so that compromise of reasoning does not automatically become compromise of action.

I do not see this as competing with identity standards. I see it as completing them operationally. Identity and delegation mechanisms establish who the agent is and what authority it may carry. A runtime governance layer determines what that authenticated, delegated agent may actually do, action by action, in real time, and preserves evidence of those decisions.

In NIST's own vocabulary, this is the Policy Decision Point / Policy Enforcement Point problem applied to AI agent actions. From that perspective, the key gap is not merely agent identity. It is whether the enterprise can prove that an agent acted within authorized bounds at the moment of execution.

Supporting technical materials, including the current reference implementation, architecture notes, audit-trail design, and mappings to NIST terminology, are available in the ZLAR repository: https://github.com/ZLAR-AI/ZLAR

I would be glad to contribute further if NCCoE advances this effort, including by sharing implementation artifacts and architecture mappings related to execution-boundary authorization, accountable delegation, and verifiable auditability for AI agents.

Attached:

1. Appendix: ZLAR Mapping to the NCCoE Concept Paper Questions
2. ZLAR Glossary

Thank you for the opportunity to comment.

Respectfully,

Vincent Nijjar
Founder, ZLAR