From Playbooks to Decisions: An Auditable Coordination Protocol for Hunter–Policy–Responder Cyber Agents (Extended Abstract)

Samuel Addington

Department of Computer Engineering and Computer Science
California State University, Long Beach
samuel.addington@csulb.edu

Abstract

While Security Orchestration, Automation, and Response (SOAR) systems operationalize incident workflows through playbooks (Cybersecurity and Infrastructure Security Agency 2021; Nelson et al. 2025), they often lack a decision-provenance (Singh, Cobbe, and Norval 2019) mechanism suited to multi-agent settings—where one agent proposes actions, another authorizes them, and a third executes them. We introduce the Cyber Agent Coordination Protocol (CACP), an auditable coordination layer that transitions agent collaboration from free-form dialogue to role-typed commitments (Singh 1999). CACP enforces explicit authority boundaries across Hunter–Policy–Responder roles and emits signed, hash-chained Decision Cards (Laurie, Langley, and Kasper 2013; Newman, Meyers, and Torres-Arias 2022) that bind (i) evidence pointers and hashes, (ii) policy checks (e.g., rules-of-engagement clauses), and (iii) execution receipts into a queryable provenance record. To avoid a centralized bottleneck, CACP supports federated logging with periodic Merkle-root checkpointing (Laurie, Langley, and Kasper 2013) to an append-only transparency log. In a small simulated testbed (50 scenarios; 2 analysts), CACP reduced mean time-to-audit by 42% (8.6s $\rightarrow <$ 4.9s) relative to unstructured agent logs, and prevented execution of out-of-policy actions in a bounded suite of 100 indirect prompt-injection attempts, while adding 22ms mean coordination latency per decision cycle (vs. 1.2s average LLM inference).

Introduction

SOAR playbooks (Cybersecurity and Infrastructure Security Agency 2021; Nelson et al. 2025) help defenders respond consistently under time pressure, but modern security operations increasingly rely on agentic workflows: LLM-based components that triage alerts, retrieve context, propose containment steps, and call tools. In these workflows, failures are less about whether a playbook step ran and more about whether an action was authorized, justified by verifiable evidence, and attributable to a specific decision pathway.

Problem (provenance gap). In a multi-agent ecosystem—e.g., a Hunter identifies anomalous behavior, a Policy component evaluates constraints, and a Responder executes containment—operators often cannot quickly answer: Which evidence justified this action? Which policy clause authorized it? Was execution performed with valid approval, or via an unsafe shortcut? Conventional logs are frequently insufficient because they (i) are not structured for audit queries, (ii) do not bind evidence→authorization→actuation, and (iii) are vulnerable to partial loss or post-hoc ambiguity.

Threat pressure. Prompt injection including indirect prompt injection delivered via retrieved content can cause agents to propose unsafe or out-of-policy actions even when the top-level task is benign. In cyber operations, "unsafe proposals" are not hypothetical; they directly map to destructive tool calls (e.g., mass file deletion, disabling telemetry, or isolating critical assets) (OWASP GenAI Security Project 2025; Greshake et al. 2023).

Approach (protocol, not chat). We shift the paradigm from "playbook execution" to accountable decision-making by treating agent coordination as a protocol. CACP constrains collaboration into typed phases with explicit authority, producing a tamper-evident record that supports rapid auditability and accountability.

Contributions:

1. **Protocol**. A role-typed coordination protocol with phases Propose/Evaluate/Commit/Rollback that expresses authority through typed, signed messages rather than free-form dialogue.
2. **Decision Card artifact**. A signed provenance object binding evidence pointers/hashes, policy checks, approvals, and execution receipts.
3. **Feasibility evidence**. A preliminary evaluation showing faster audit queries and bounded suppression of out-of-policy execution with negligible overhead relative to LLM inference.

The CACP Protocol

CACP decomposes cyber operations into three specialized roles, enforcing separation-of-duties:

- The Hunter (H): Scans telemetry to generate hypotheses and produces evidence pointers (ptr_{evid}) but lacks execution authority.
- The Policy Agent (P): The "constitutional" layer. It evaluates proposals against a risk budget and RoE, providing

a signed cryptographic grant ($auth_{sig}$). To ensure scalability, the Policy Agent operates as a reference-monitor-style authorization gate, where LLM-based semantic reasoning is verified against static RoE schemas before approval (Saltzer and Schroeder 1975). RoE checks are enforced by a deterministic schema/rule engine; the LLM may propose interpretations but cannot approve a grant.

- The Responder (R): The execution layer. It only interacts with the environment if presented with a valid, signed proposal from both H and P.
- The Federated Ledger (L): Instead of a centralized bottleneck, local agent clusters maintain high-speed logs that are periodically "checkpointed" to a global, immutable ledger using Merkle Tree summary hashes.

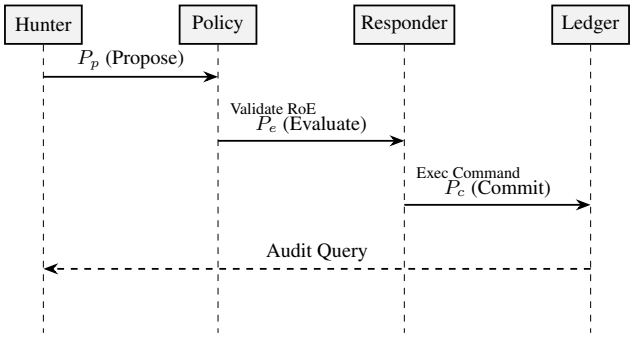

Figure 1: The CACP Coordination Cycle showing role-typed transitions (P_p, P_e, P_c) required for non-repudiable decision provenance.

Formal Grammar and Decision Cards

The protocol follows a strictly typed message grammar to reduce role confusion and unauthorized actuation by requiring explicit, verifiable authorization tokens before execution. Every successful coordination cycle culminates in a Decision Card. This object serves as a non-repudiable record.

Threat Model & Security Properties

We assume (i) a Dolev–Yao network adversary (Dolev and Yao 1983) who can observe, replay, and reorder messages, and (ii) a single-role compromise model where any one of H, P, or R may be manipulated (e.g., via malware or prompt injection). We explicitly consider indirect prompt injection (OWASP GenAI Security Project 2025; Greshake et al. 2023), where retrieved content contains adversarial instructions that steer tool-using agents.

CACP does not claim to "solve prompt injection," guarantee correct policy judgment, or prevent all malicious behavior. Its goal is to:

1. Gate execution on explicit authorization: It only interacts with the environment if presented with a valid Policy grant tied to a specific proposal (and optionally a Hunter signature for attribution).
2. Bind evidence→policy→action: Decision Cards cryptographically link evidence hashes/pointers, policy clauses, and execution receipts.

```
{
  "decision_card_id": "dc-2026-01-04-0014",
  "task_id": "task-7781",
  "provenance": {
    "hunter": {
      "hypothesis": "SMB lateral movement",
      "evidence_ptr": "log://clusterA/evid/99182",
      "evidence_hash": "sha256:e3b0c442... (truncated
          )"
    },
    "policy": {
      "roe_rule_id": "ROE-SEC-04",
      "risk_score": 0.87,
      "auth_sig": "sig:8f2a3c... (truncated)"
    },
    "responder": {
      "action": "isolate_endpoint",
      "execution_receipt_sig": "sig:31ac09... (
          truncated)",
      "delta_state": {
        "edr": "host quarantined",
        "firewall": "egress restricted"
      }
    }
  }
}
```

Listing 1: Sample CACP Decision Card Object.

3. Make tampering detectable: hash chaining and append-only logging make deletion/modification detectable under ledger assumptions.

Preliminary Evaluation

We conducted a preliminary evaluation using a simulated testbed with LLM-based agents with tool-calling, 50 attack scenarios, and two analysts performing audit tasks. Each scenario produced one action decision; analysts answered two fixed audit queries (evidence and RoE clause) using a standardized form. MTTA was measured from log/DecisionCard open to correct answer submission; we report means over scenarios.

Baselines

We measured the time required for a human analyst to determine the "Root Cause of Decision" using CACP Decision Cards versus standard unstructured agent logs.

- B0: Unstructured logs — chat-style logs + timestamps.
- B1: Structured logs (no commitments) — same fields as Decision Cards (evidence ptr, RoE clause, action) but no signatures, no phase gating, no hash chaining.
- CACP — typed phases + Policy grants + signed Decision Cards + commit receipts.

Metrics

- Mean Time to Audit (MTTA): time to answer "what evidence and policy clause justified this action?"
- Out-of-policy execution rate: fraction of injection-induced proposals that lead to execution (bounded suite).

Phase	Transition	Formal Payload Definition (P)	Security Property
Propose	$H \to P$	$P_p = \{ID_{task}, ptr_{evid}, h_{evid}, cmd_{prop}, risk_{est}\}$	Evidence Pinning
Evaluate	$P \to R$	$P_e = \{ID_{task}, score_{risk}, clause_{ROE}, res \in \{acc, rej\}, \sigma_P\}$	Policy Enforcement
Commit	$R \to L$	$P_c = \{ID_{task}, \Delta_{state}, \sigma_R(P_p \cup P_e), hash_{prev}\}$	Non-Repudiation
Rollback	$R \to L, H$	$P_r = \{ID_{task}, error_{code}, sig_{rollback}, hash_{prev}\}$	Fault Tolerance

Table 1: CACP Formal Message Grammar: Role-Based Transitions and Cryptographic Commitments.

- Coordination latency: additional time for Propose→Authorize→Commit.

Results

- Audit-query answerability: CACP reduced MTTA by 42% (8.6s → 4.9s) vs B0, and improved over B1 (structured logs alone).
- Policy-violation prevention (bounded): Under 100 indirect prompt-injection attempts (OWASP GenAI Security Project 2025; Greshake et al. 2023) targeting the Hunter, CACP prevented execution of out-of-policy actions in this test suite by requiring an RoE-matching Policy grant prior to execution. Out-of-policy means the proposed action scope violates the RoE clause referenced in P_e (Table 1).
- Overhead: mean coordination latency 22ms per decision cycle, negligible relative to 1.2s LLM inference latency observed in our environment.

Limitations. Small-N and synthetic scenarios; broader evaluation should include diverse incidents, more attackers, and compromise of each role (especially Policy), plus ledger consistency monitoring.

Related Work

SOAR playbooks provide operational structure, but typical audit trails record events without cryptographically binding the decision path across multiple autonomous components. Work on prompt injection (OWASP GenAI Security Project 2025; Greshake et al. 2023) (including indirect prompt injection) shows how retrieved content can steer tool-using systems toward unsafe behaviors. Separately, transparency logs (Laurie, Langley, and Kasper 2013) and Merkle-tree append-only logging demonstrate how to make post-hoc tampering detectable via inclusion and consistency proofs. CACP connects these threads by applying protocol discipline and transparency-inspired logging to cyber decision provenance (Singh, Cobbe, and Norval 2019), emphasizing operational accountability rather than model interpretability.

Discussion and Future Work

Scaling to swarms

The Hunter–Policy–Responder triad can serve as a unit of composition. Future work will explore hierarchical coordination, where lead Policy agents aggregate Decision Cards across sub-swarms, with federated audit ledgers to avoid bottlenecks.

Ledger consistency and monitoring

To handle equivocation and forks, future deployments should incorporate transparency-style monitoring and consistency proofs, and define explicit governance for who can append and checkpoint.

Resilience to covert channels

Enforcing the strict message grammar reduces free-form channels where hidden instructions may reside; additional work is needed to detect covert side channels and collusion strategies.

Dynamic policy adaptation

In real operations, policies change quickly. We plan versioned policy schemas and explicit policy provenance embedded in Decision Cards to support post-incident reasoning over policy evolution.

Ethics and Broader Impact

The development of the Cyber Agent Coordination Protocol (CACP) aims to increase the transparency and accountability of autonomous systems in high-stakes environments. By enforcing role-typed commitments and generating signed Decision Cards, our work directly addresses the ethical risk of "unaccountable autonomy," where the reasoning behind defensive actions becomes opaque to human oversight.

Conclusion

This paper has presented the Cyber Agent Coordination Protocol (CACP), a framework designed to bring accountability and auditability to autonomous multi-agent cyber defense. By moving beyond static playbooks and free-form agent dialogue, CACP establishes a formal mechanism for decision provenance. Through the use of role-typed commitments and tamper-evident Decision Cards, we provide a practical bridge between high-speed autonomous operations and the critical need for human oversight on the cyber battlefield.

Our preliminary evaluations demonstrate that while structured coordination introduces a minor latency trade-off, the resulting gains in policy enforcement and forensic clarity are indispensable for deploying AI agents in high-stakes, adversarial environments. As the "Cyber Battlefield" grows in complexity, protocols like CACP will be foundational in ensuring that autonomous systems remain not only effective but—most importantly—accountable.

References

Cybersecurity and Infrastructure Security Agency. 2021. Federal Government Cybersecurity Incident and Vulnerability Response Playbooks. https://www.cisa.gov/resources-tools/resources/incident-and-vulnerability-response-playbooks. Accessed: 2026-01-24.

Dolev, D.; and Yao, A. C. 1983. On the Security of Public Key Protocols. *IEEE Transactions on Information Theory*, 29(2): 198–208.

Greshake, K.; Abdelnabi, S.; Mishra, S.; Endres, C.; Holz, T.; and Fritz, M. 2023. Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. arXiv:2302.12173.

Laurie, B.; Langley, A.; and Kasper, E. 2013. Certificate Transparency. RFC 6962.

Nelson, A.; Rekhi, S.; Souppaya, M.; and Scarfone, K. 2025. Incident Response Recommendations and Considerations for Cybersecurity Risk Management: A CSF 2.0 Community Profile. NIST Special Publication 800-61r3, National Institute of Standards and Technology.

Newman, Z.; Meyers, J. S.; and Torres-Arias, S. 2022. Sigstore: Software Signing for Everybody. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*.

OWASP GenAI Security Project. 2025. LLM01: Prompt Injection. https://genai.owasp.org/llmtop10-2025/llm01-prompt-injection/. Accessed: 2026-01-24.

Saltzer, J. H.; and Schroeder, M. D. 1975. The Protection of Information in Computer Systems. *Proceedings of the IEEE*, 63(9): 1278–1308.

Singh, J.; Cobbe, J.; and Norval, C. 2019. Decision Provenance: Harnessing Data Flow for Accountable Systems. *IEEE Access*, 7: 6562–6574.

Singh, M. P. 1999. An Ontology for Commitments in Multiagent Systems. *Artificial Intelligence and Law*, 7(1): 97–113.