Beyond Prompt Filtering: A Survey on Action-Level Safety Enforcement for Multi-Agent LLM Systems

Anonymous ACL submission

Abstract

Large language model (LLM) agents increasingly execute real-world actions—calling APIs, modifying databases, and triggering workflows—yet the guardrail literature still emphasizes prompt- and response-level moderation. This paper synthesizes *action-level safety enforcement* for multi-agent LLM systems. We organize the literature through a Horizon–Modality lens that separates *where* enforcement observes behavior (single actions, full trajectories, or federated multi-agent workflows) from *what type of evidence* it uses (symbolic rules, information-flow tracking, or empirical monitoring). Across 62 primary sources spanning peer-reviewed venues, preprints, and selected industry reports, we identify an underdeveloped evaluation profile: non-adversarial, compositional compliance risks that arise from organizational boundaries, context fragmentation, and provenance loss across agent handoffs. We use compliance-fragmentation risks and semantic laundering as prospective analytical lenses and chart open problems in federated verification, provenance preservation, and benchmark design.

1 Introduction

LLM-based agents have moved beyond text generation into autonomous action execution. A 2026 Cloud Security Alliance survey (N=418) found that 82% of enterprises have unknown AI agents in their infrastructure, while 65% reported agent-related security incidents in the past year (Cloud Security Alliance and Token Security, 2026). A parallel Gravitee study (N=919) revealed that only 47% of deployed agents receive any security monitoring (Gravitee, 2026). The EU AI Act's high-risk obligations take effect in August 2026 (European Parliament and Council, 2024), and OWASP released a dedicated Top 10 for Agentic Applications in December 2025 (OWASP GenAI Security Project, 2025), signaling that the threat model has shifted from *what the model says* to *what the agent does*—a guardrail failure in 2024 meant a bad response; in 2026 it can mean data deleted, funds transferred, or privileges escalated.

The prompt-filtering gap. Existing guardrail surveys emphasize prompt-level defenses: jailbreak detection (Wang et al., 2025b), content moderation (Dong et al., 2025), and alignment training (Kumar et al., 2025). These do not govern the intermediate *actions* an agent takes—tool invocations, database writes, or cross-agent delegations. Chen et al. (2026) report that specialized guardrails achieve below-20% rejection rates on unsafe multi-step trajectories, illustrating that surface-level inspection misses action-sequence violations.

Non-adversarial compliance risks. Most agent-safety research assumes an adversarial threat model. We highlight a complementary class of risk: situations where every agent appears locally compliant, yet the composite workflow may violate a global constraint that no single agent can detect. We call this pattern *compliance-fragmentation risk* and use it as an analytical lens throughout this survey. No public incident report has yet documented a confirmed instance; rather, it is a prospective model motivated by partial observability, heterogeneous policies, and provenance loss across LLM-mediated handoffs—echoing "normal accidents" in tightly coupled systems (Perrow, 1984). Indirect signals are consistent: MAST's analysis of 1,600+ traces identifies *information withholding* (11.3%) and *context inconsistency* (9.8%) (Cemri et al., 2025), though framed as performance failures rather than compliance violations.

Contributions. This survey makes three contributions to this emerging direction:

1. A **Horizon–Modality organizing lens** (Figure 1) that maps defenses along two dimensions: the *scope* of safety enforcement (action / trajec-

tory / federated) and the *modality* of guarantee (symbolic / information-flow / empirical).

2. A **cross-community synthesis** that unifies techniques from runtime verification (Sánchez et al., 2018; Zhang et al., 2025), information-flow control (Sabelfeld and Myers, 2003; Costa et al., 2025), and AI guardrails engineering (Rebedea et al., 2023; Inan et al., 2023) into one landscape.
3. A **literature-grounded gap analysis** that identifies non-adversarial, compositional compliance risks—using compliance-fragmentation risks and semantic laundering as prospective analytical lenses—as an underexplored area that current benchmarks and enforcement systems do not systematically evaluate.

Table 1 positions this survey relative to existing work.

Survey methodology. We surveyed papers and technical reports through April 2026 from ACL Anthology, arXiv, ACM/IEEE venues, and security proceedings, using queries combining terms such as *"LLM agent safety," "runtime verification agent," "information flow LLM," "multi-agent compliance,"* and *"guardrail tool-use,"* supplemented by forward and backward citation chasing. We included work that (i) enforces or evaluates safety at the level of tool calls, execution traces, or inter-agent handoffs; (ii) introduces runtime verification, information-flow, or empirical monitoring for LLM agents; or (iii) provides benchmarks, taxonomies, or governance evidence for agentic safety. Work focused solely on static prompt moderation or non-agentic text generation was excluded. This process yielded 62 primary sources from an initial pool of approximately 400 candidate documents. Of these, 34 are peer-reviewed (published at venues such as ICLR, NeurIPS, EMNLP, ICSE, and CAV), 24 are arXiv preprints, and 4 are technical reports or industry white papers, including three industry surveys and one regulatory framework document. All sources are English-language. Systems were coded along Horizon–Modality dimensions by the authors; we distinguish peer-reviewed from preprint work where relevant. We do not claim PRISMA-style exhaustive coverage.

Positioning relative to related work. Table 1 compares this survey against four threads. *Mind the Agent* (Zhou et al., 2025) and the SoK on guardrails (Wang et al., 2025b) cover adversarial attacks and prompt-level defenses but do not address runtime verification or information-flow control. MAST (Cemri et al., 2025) provides the first empirical multi-agent failure analysis (14 modes, κ=0.88) but frames failures as performance issues, not compliance violations. The Agent Harness survey (He et al., 2026) treats execution infrastructure but not safety enforcement. Our work addresses the intersection: action-level enforcement, non-adversarial compliance risks, and federated multi-agent scope.

2 Background and Organizing Framework

2.1 Agent Architecture

We adopt the standard definition: an LLM agent $\mathcal{A} = (f_\theta, \mathcal{T}, \mathcal{M}, \mathcal{P})$ comprises a language model f_θ, a tool set $\mathcal{T}$, a memory $\mathcal{M}$ (including short-term context and long-term retrieval stores), and a local policy $\mathcal{P}$ (Wang et al., 2024). In a MAS, the memory $\mathcal{M}_i$ of each agent $\mathcal{A}_i$ is private and incomplete: no single agent holds the full execution context. A *multi-agent system* (MAS) consists of agents $\{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$ communicating through tool-use and agent-to-agent protocols such as MCP (Acharya and Gupta, 2026). Each agent operates under a local policy P_i, while a global compliance requirement P_{global} constrains the composite workflow.

2.2 Key Concepts

Compliance-fragmentation risks. A *compliance-fragmentation risk* arises when each agent satisfies its local policy yet the composite workflow may violate a global constraint undetectable by any single agent—a specific instance of the compositional verification problem (McMillan, 1999) applied to the LLM-agent setting. This is an *analytical lens*, not a formally established category; see Appendix B and §5.

Semantic laundering. When information flows between agents via LLM-mediated reformulation, taint labels attached to source data may be lost—a phenomenon we call *semantic laundering*, drawing on classical implicit-flow observations (Denning, 1976; Sabelfeld and Myers, 2003). We use a distinct term rather than "implicit flow" because the mechanism differs: classical implicit flows leak information through deterministic control-flow branches, whereas LLM reformulation loses *provenance metadata* through stochastic, many-to-many token mixing under dense self-attention (Siddiqui

Dimension	Mind the Agent (Zhou et al., 2025)	SoK Guardrails (Wang et al., 2025b)	MAST (Cemri et al., 2025)	Harness Survey (He et al., 2026)	This Survey
Adversarial threat model	✓	✓	○	○	○
Non-adversarial compliance	✗	✗	○	✗	✓
Action-level enforcement	○	✗	✗	○	✓
Multi-agent scope	○	✗	✓	○	✓
Runtime verification (RV)	✗	✗	✗	○	✓
Information-flow / taint	✗	✗	✗	✗	✓
Benchmark gap analysis	✗	○	✗	✗	✓

Table 1: Positioning of this survey relative to four representative threads: two safety surveys (Mind the Agent, SoK Guardrails), one failure taxonomy (MAST), and one systems survey (Agent Harness). ✓ = primary focus, ○ = partial coverage, ✗ = not a primary focus in the stated scope. All comparative dimensions are drawn from established concepts in the literature (e.g., adversarial threat models, runtime verification, information-flow analysis). In addition to these shared dimensions, our survey uses compliance-fragmentation risks and semantic laundering as *prospective analytical lenses* for identifying underexplored research directions (§5); we do not include these as comparative dimensions because they are observational categories introduced by this work rather than independently established concepts.

et al., 2024). The result is not a covert channel but the destruction of the labels themselves—a phenomenon closer to declassification without authorization than to information leakage. Formalizing the resulting mutual-information dilution remains open.

2.3 Horizon–Modality Organizing Lens

We organize the literature along two dimensions (Figure 1): *where* enforcement observes behavior, and *what type of evidence* it uses. This lens is intentionally coarse-grained—an organizing device, not a formal taxonomy.

Horizon captures the *scope* of a defense:

- **Action**: checks a single tool call or API invocation.
- **Trajectory**: monitors a sequence of actions within one agent session, detecting violations that emerge only over multiple steps.
- **Federated**: coordinates safety checks *across* agent boundaries in a MAS, addressing distributed compliance and compositional risks.

Modality captures the *type of evidence* used to assess safety:

- **Symbolic**: rule-based, formally specified (e.g., temporal logic, policy DSLs). Provides verifiable guarantees under explicit and correctly specified policies.
- **Information-Flow**: tracks data provenance via taint labels (Sabelfeld and Myers, 2003). IFC belongs to the formal-methods family but addresses data provenance rather than behavioral constraints, motivating a separate column.
- **Empirical**: learned classifiers, LLM-as-judge, simulation-based testing. Most flexible but lacks the formal guarantees of the above categories.

This three-way distinction is an *organizational convenience*, not a guarantee-strength ordering; symbolic and IFC are peers in the formal-methods tradition, separated here because they address orthogonal concerns (behavioral vs. data-flow properties). We place them on the same axis because practitioners must choose *which type of evidence* to instrument at each enforcement point, making the choice directly comparable even though the underlying theories are independent. An alternative design could nest IFC under Symbolic; we opt for a flat structure because the agent-safety literature treats taint tracking and rule-based monitoring as distinct engineering choices with different integration costs.

The 3×3 matrix yields nine cells; every surveyed system maps to at least one (Table 2). Density concentrates in Action–Symbolic and Action–Empirical, while the Federated row remains sparsely populated in our surveyed corpus. This diagonal density gradient—from well-explored to nearly vacant—provides a structural roadmap for the field.

Table 3 presents our four-category risk taxonomy for action-level safety, organized by the nature of harm in the agent's action space rather than by attack vector.

3 Pre-Execution Defenses

We cover pre-execution defenses briefly; the primary contribution lies in runtime (§4) and federated (§5) enforcement.

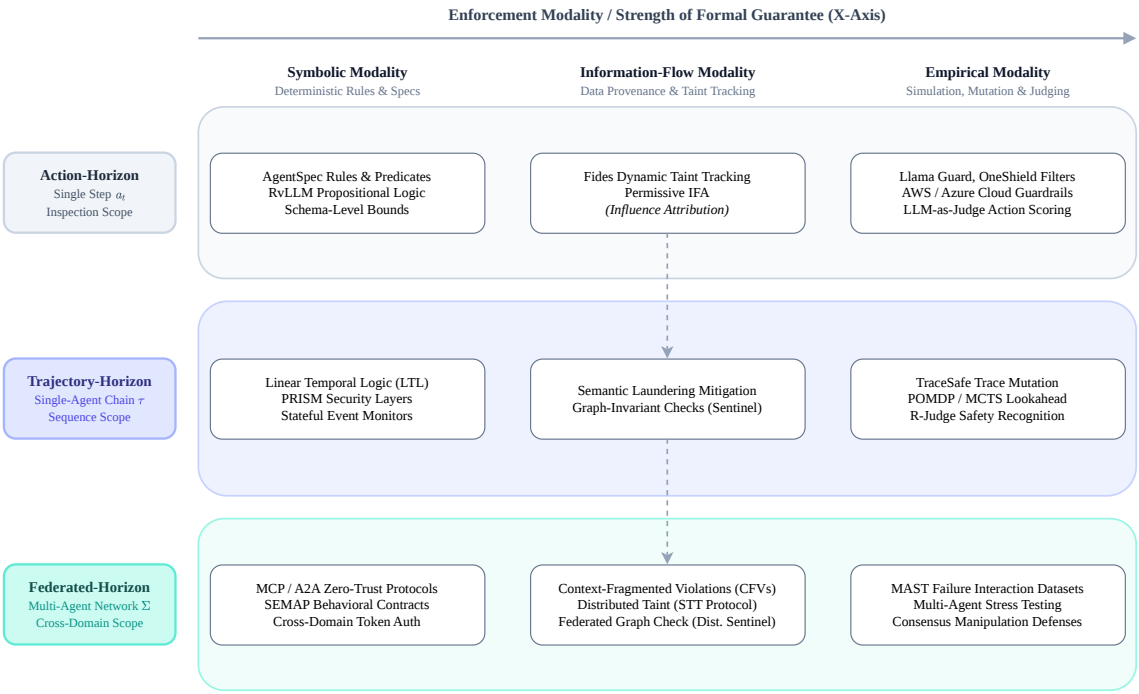

Figure 1: The Horizon–Modality organizing lens. Rows represent the *scope* of safety enforcement (Action, Trajectory, Federated); columns represent the *type of evidence* used (Symbolic, Information-Flow, Empirical). Each cell lists representative systems. Density decreases from Action–Symbolic (most populated) to Federated–IFC (least populated), suggesting that broader-scope defenses remain underexplored. Note: Symbolic and IFC are both grounded in formal methods; we separate them because they address orthogonal concerns (behavioral properties vs. data-flow properties).

Policy Specification and Capability Bounding. AgentSpec (Wang et al., 2026a) provides a DSL for runtime constraints as predicate-trigger-enforcement triples, effective across code execution and embodied agents but limited to single-agent settings. Policy-as-Prompt (Kholkar and Ahuja, 2025) automates translation of design documents into guardrail classifiers. Capability bounding (Li et al., 2007) restricts an agent's action space *a priori* via schema-level tool filtering (He et al., 2026), task-scoped tokens (OWASP GenAI Security Project, 2025), sandboxed execution (PRISM (Li, 2026), gVisor (Young et al., 2019)), and layered frameworks such as TrustAgent (Hua et al., 2024).

Planning-Phase Safety. Forward simulation via POMDP (Silver et al., 2017) and MCTS (Browne et al., 2012) prunes unsafe action branches but remains single-agent with explicit state spaces. Superego (Watson et al., 2025) layers a constitutional oversight agent that evaluates actions against value principles before execution.

4 Runtime Enforcement

Pre-execution defenses establish boundaries, but many violations only become detectable *during* execution—when behavior diverges from expected patterns or information flows across unanticipated boundaries.

4.1 Symbolic Runtime Verification

RvLLM (Zhang et al., 2025) introduces ESL, combining natural language with propositional logic for runtime constraint verification, increasing true-positive rates by 15–50% on legal violation detection with modest specification effort (31 regulations in three person-days).

For multi-step trajectories, graph-invariant monitoring and PDG-based trace abstraction (Wang et al., 2025a) detect *policy-invisible violations*—breaches that appear compliant step-by-step but violate structural invariants (e.g., information-flow paths, privilege escalation chains) over the full action trace.

Table 2: Horizontal comparison of action-level enforcement mechanisms. **H/M**: Horizon–Modality cell. •=low latency, ••=medium, • • •=high. ✓=yes, ○=partial, ✗=no.

System	H/M Cell	Enforcement Granularity	Latency	Multi-Agent	Stateful	Formal
AgentSpec	Act/Sym	Predicate per tool call	•	✗	✗	✓
RvLLM	Act–Traj/Sym	Propositional logic + forward chaining	•	✗	✓	✓
Fides	Act/IFC	Taint label per data object	••	✗	✓	✓
Permissive IFA	Act/IFC	Token-level attribution	••	✗	✓	○
Graph-Invariant Monitor	Traj/IFC	Structural invariant per trace	••	✗	✓	✓
Distributed Taint (STT)	Fed/IFC	Cross-agent taint propagation	• • •	✓	✓	○
TraceSafe	Traj/Emp	LLM judge per trajectory	• • •	✗	✓	✗

Table 3: Taxonomy of action-level safety risks in multi-agent LLM systems. Each category is characterized by the nature of the resulting harm, whether adversarial intent is required, the minimum agent topology in which the risk arises, and the primary enforcement mechanisms that address it (section references in parentheses).

Risk Category	Illustrative Scenario	Min. Topo.	Detect.	Primary Defenses
Unauthorized Data Egress	Agent sends PII to external API; summary encodes classified content	Single	Per-action	IFC / taint tracking (§4.2), sandboxing (§3)
Policy-Violating Tool Invoc.	DB write during read-only audit; transaction w/o dual-auth	Single	Per-action	AgentSpec predicates (§3), RV (§4.1)
Cascading Compliance Fail.	Multi-step workflow violates HIPAA minimum necessary standard	Single*	Multi-step	Trajectory analysis (§4.3), graph invariants (§4.1)
Emergent Composition Viol.	Cross-domain agents collectively breach sanctions; consensus manipulation	Multi	Cross-agent	Distributed taint (§4.2), federated verif. (§5.2)

*Can occur in single-agent multi-step workflows but is amplified in multi-agent settings.

Note: Categories are not mutually exclusive; a single incident may span multiple categories.

4.2 Information-Flow Control

IFC tracks data provenance via taint labels with deterministic propagation rules (Sabelfeld and Myers, 2003; Li et al., 2002). **Fides** (Costa et al., 2025) applies dynamic taint tracking to LLM agent planners with *controlled revelation*—selective hiding that balances security and utility. Its design trade-offs define the current frontier: single-agent scope, explicit secrecy rather than full non-interference, and manual label definitions. Crucially, Fides does not address semantic laundering: LLM paraphrasing washes provenance labels through stochastic transformations qualitatively different from classical instruction-level taint propagation (Sabelfeld and Myers, 2003).

Permissive IFA (Siddiqui et al., 2024) addresses over-tainting by propagating only labels of inputs that *causally influenced* the output via token-level attribution. Cross-agent extensions propagate taint labels across agent boundaries in federated settings.

4.3 Empirical Runtime Monitoring

TraceSafe (Chen et al., 2026) mutates benign traces into harmful variants, revealing that specialized guardrails achieve below-20% rejection on unsafe trajectories while general-purpose LLMs show hyper-sensitive 98–100% rejection on both safe and unsafe traces. The No Free Lunch hypothesis (Kumar et al., 2025) formalizes this tension: strengthening security necessarily degrades utility or usability.

Systemic limitations of empirical monitoring. If the monitor itself is an LLM, it is subject to the same adversarial pressures as the agent it supervises—multi-step query decomposition can bypass LLM-based guardrails by splitting harmful requests into innocuous sub-queries. This arms-race dynamic motivates layered architectures that combine empirical monitors with symbolic or IFC-based backstops, mirroring defense-in-depth principles. Figure 1 reflects this insight: the most robust enforcement stacks span multiple Modality columns at each Horizon level, rather than relying

on a single modality.

Runtime defenses exhibit a three-way trade-off: symbolic monitors offer guarantees under explicit policies, IFC captures provenance violations, and empirical monitors scale but provide weaker assurance—motivating hybrid stacks. A complete classification appears in Appendix Table 5.

4.4 Language-Specific Phenomena and Safety Enforcement

A distinctive challenge is that the "programs" being monitored are stochastic language models shaped by linguistic phenomena. We highlight four under-explored interactions that affect all three horizons.

Tokenization and Taint Boundaries. Subword tokenizers (BPE, WordPiece) split sensitive entities (e.g., `patient_SSN` → `patient+_SS+N`) across token boundaries, causing word-level taint labels to fragment. We are not aware of an agent-oriented IFC system that treats subword tokenization as a first-class taint-boundary problem.

Attention Mechanisms and Semantic Laundering. Under dense self-attention, every output token mixes all inputs, causing a single sensitive input to contaminate the entire output (Siddiqui et al., 2024). Sparse attention patterns suggest not all heads propagate information equally, motivating *head-level taint tracking*—an unexplored direction. Similarly, decoding parameters (temperature, top-k) affect violation distributions but no enforcement system accounts for them.

In-Context Learning as Compliance Bypass. In-context learning (ICL) enables agents to acquire behavioral patterns from few-shot examples (Ouyang et al., 2022), creating compliance risks: an agent enforcing HIPAA can be steered by non-compliant in-context examples. Unlike prompt injection, ICL bypasses exploit the model's learning mechanism, evading pattern-based filters.

RLHF Alignment and Runtime Verification. RLHF (Ouyang et al., 2022) and Constitutional AI (Bai et al., 2022) internalize safety preferences, but red-teaming (Zou et al., 2023; Ganguli et al., 2022; Perez et al., 2023) shows these guarantees do not transfer to agentic tool-use settings. Runtime verification complements alignment: training reduces violation *frequency*; enforcement provides *guarantees*. Probabilistic approaches such as ProbGuard (Wang et al., 2026b) learn DTMCs from traces to predict violations proactively. Whether RLHF-aligned models produce more monitorable behavior remains open.

5 Multi-Agent Safety Challenges

This section covers the Federated row in Figure 1.

5.1 Why Single-Agent Safety Does Not Compose

MAST (Cemri et al., 2025) identifies 14 failure modes across 1,600+ traces, clustered into specification (37%), inter-agent misalignment (34%), and verification failures (29%)—but focuses on task performance, not compliance. A workflow can succeed functionally while violating data-handling rules invisible to any single agent.

Sources of non-compositionality. Single-agent safety does not compose for five reasons: **partial observability** (each agent sees only its local context); **heterogeneous policies** (different local rules make global policy a reconciliation problem); **provenance loss** (LLM-mediated handoffs drop security labels); **temporal drift** (compliance conditions change between validation and action); and **accountability fragmentation** (no single agent acts unsafely, yet the workflow violates a cross-domain constraint).

Illustrative scenarios. Two hypothetical scenarios (not documented incidents) illustrate these risks. *Healthcare*: agents $\mathcal{A}_1$–$\mathcal{A}_3$ extract, aggregate, and publish patient data; each satisfies its local policy, but the composite workflow could violate HIPAA's minimum-necessary standard. *Financial*: a procurement agent validates a vendor against sanctions lists, but a temporal gap before payment means the composite workflow could execute a prohibited transaction. In both cases, violation-relevant information is distributed across agents and LLM-mediated communication does not preserve policy metadata. Whether such patterns occur at meaningful rates remains open (Cemri et al., 2025). However, rapid MCP growth (177K tools, "action" tools rising from 27% to 65% in two years (Stein, 2026)) suggests these risks will intensify as multi-agent deployments scale.

5.2 Distributed Compliance Verification

Distributed compliance draws on assume-guarantee reasoning (McMillan, 1999) and Byzantine fault tolerance (Lamport et al., 1982). McMillan's circular compositional rule—originally developed

for hardware verification—is domain-independent at its core: it shows that if component A is correct assuming B's guarantee, and vice versa, then the composed system satisfies both guarantees. This abstract structure applies whenever independently verified modules interact, including LLM agents with local compliance policies. However, neither framework was designed for stochastic, natural-language-mediated agents. Adapting these foundations—where outputs are probabilistic and handoffs are mediated by lossy summaries—remains an open challenge. Recent work explores structural taint tracking via sidecar monitors at agent boundaries, propagating provenance labels alongside inter-agent messages so that downstream agents can verify whether upstream data handling satisfied relevant policies before acting on it.

5.3 Agent-to-Agent Protocol Security

Federated compliance depends on the security of the communication substrate. A formal MCP security framework (Acharya and Gupta, 2026) identifies 23 threat vectors; an empirical study of 177,436 MCP tools found "action" tools rose from 27% to 65% between 2024 and 2026 (Stein, 2026). These findings suggest that cross-protocol interoperability raises unresolved trust-model questions. SEMAP (Mao et al., 2025) proposes behavioral contracts inspired by assume-guarantee reasoning (McMillan, 1999) to reduce misalignment failures.

These challenges explain why the Federated $\times$ IFC cell remains least populated—an underdeveloped area, not proof that compositional failures are widespread.

6 Evaluation Landscape

Table 4 summarizes the benchmark landscape. Most benchmarks evaluate adversarial robustness (AgentHarm (Andriushchenko et al., 2025), AgentSafetyBench (Zhang et al., 2024)) or multi-step trajectory interception (TraceSafe (Chen et al., 2026), OS-Harm (Kuntz et al., 2025)). R-Judge (Yuan et al., 2024) finds frontier models miss 40% of risks human annotators detect. MAST (Cemri et al., 2025) offers the only multi-agent dataset (1,600+ traces) but focuses on task performance, not compliance.

The compliance gap. No benchmark we surveyed evaluates *non-adversarial compliance violations* in multi-agent settings. EU-AgentBench (Lichkovski et al., 2025) approaches legal compliance but remains single-agent. The field also lacks standardized metrics capturing the rate at which compliant individual actions compose into non-compliant workflows. Three self-reported industry surveys (Appendix Table 7, N=1,637) report 65–88% of organizations experienced agent-related incidents (Cloud Security Alliance and Token Security, 2026; Gravitee, 2026; Arkose Labs, 2026). These figures should be interpreted cautiously: the surveying organizations sell security products, respondents are self-selected, and comparable incident rates for non-agent enterprise automation are not publicly available as a baseline. We treat these as directional evidence of governance pressure, not direct measurements of compliance-fragmentation risk.

What a compositional compliance benchmark would require. Such a benchmark would need distributed multi-agent workflows with global policies not reducible to local allow/deny rules, provenance-rich execution traces that distinguish action-choice from information-flow violations, and benign-intent scenarios where each local action appears permissible yet the composite workflow violates a cross-agent constraint. These requirements explain why extending single-agent harmful-task benchmarks is insufficient.

7 Open Problems

The Horizon–Modality lens reveals four structurally underexplored directions:

OP1: Federated Verification at Scale (Federated $\times$ all). The Federated row is sparsely populated. Scaling to production MAS requires adapting compositional verification (McMillan, 1999) for non-deterministic LLM behavior—likely through probabilistic assume-guarantee reasoning.

OP2: Provenance Preservation (Traj/Fed $\times$ IFC). Permissive IFA (Siddiqui et al., 2024) reduces over-tainting within single calls, but multi-hop provenance loss—where policy-relevant metadata degrades through successive LLM reformulations across agents—remains unsolved. Under dense self-attention, every output token mixes all inputs, making classical instruction-level taint propagation inapplicable. A solution likely requires token-level attribution combined with cross-agent provenance graphs, though formalizing "policy-relevant content preservation" across semantic transformations

Benchmark	Venue	Scale	Capability Coverage					H/M Cell
			Multi-step	Tool Use	Non-adv.	Multi-agent	Compliance	
AgentHarm (Andriushchenko et al., 2025)	ICLR'25	440	✓	✓	✗	✗	✗	Act/Emp
Agent-SafetyBench (Zhang et al., 2024)	arXiv'24	2,000	✓	✓	○	✗	✗	Act/Emp
ToolEmu (Ruan et al., 2023)	ICLR'24	144	✓	✓	✓	✗	✗	Act/Emp
TraceSafe (Chen et al., 2026)	arXiv'26	900+	✓	✓	✗	✗	✗	Traj/Emp
EU-Agent-Bench (Lichkovski et al., 2025)	arXiv'25	200+	✓	✓	✓	✗	○	Act/Sym
OS-Harm (Kuntz et al., 2025)	NeurIPS'25	300+	✓	✓	✗	✗	✗	Traj/Emp
SafeAgentBench (Yin et al., 2024)	arXiv'24	750	✓	○	✓	○	✗	Act/Emp
R-Judge (Yuan et al., 2024)	EMNLP'24	162	✓	✓	○	✗	✗	Traj/Emp
MAST-Data (Cemri et al., 2025)	NeurIPS'25	1,600+	✓	✓	✓	✓	✗	Fed/Emp
Needed (compliance bench.)	—	—	✓	✓	✓	✓	✓	*Fed/all*

Table 4: Agent safety benchmarks. Non-adv.=non-adversarial intent. ✓=yes, ○=partial, ✗=no. The *Needed* row shows the profile of a benchmark targeting non-adversarial federated compliance—currently absent from the literature.

is itself open.

OP3: Compliance Metrics. Benchmarks measure binary pass/fail on individual actions; compositional compliance requires cross-agent metrics. Proposed metrics for compositional compliance remain preliminary and lack community-wide validation; the field first needs empirical studies establishing whether such failures occur at meaningful rates in deployed systems.

OP4: Neuro-Symbolic Bridges (diagonal cells). Symbolic approaches require manual specification; empirical ones lack guarantees. Neuro-symbolic integration (Hakim et al., 2026)—using learned components to *discover* specifications enforced symbolically—is promising but unexplored for agent safety. Early results in cybersecurity suggest structured architectures outperform single-agent approaches, motivating adaptation to compliance verification.

8 Conclusion

This survey organized action-level safety enforcement for multi-agent LLM systems through a Horizon–Modality lens. Symbolic verification, information-flow control, and empirical monitoring offer complementary strengths, but federated compliance verification, provenance preservation, and non-adversarial benchmarks remain underdeveloped. Establishing the empirical prevalence of compliance-fragmentation risks is itself a pressing open problem. We hope this synthesis provides a shared vocabulary for multi-agent systems whose safety composes across organizational boundaries.

Limitations

This work has several limitations. The field evolves rapidly (cutoff: April 2026). The Horizon–Modality lens is an organizing convenience, not a formal taxonomy. Industry data (Cloud Security Alliance and Token Security, 2026; Gravitee, 2026; Arkose Labs, 2026) is self-reported. Compliance-fragmentation risks and semantic laundering are prospective categories whose empirical prevalence remains unestablished. Coverage focuses on text-based, English-language LLM agents; cross-lingual compliance and embodied agents are not addressed. We do not claim PRISMA-style exhaustive coverage.

Ethics Statement

This paper synthesizes existing published research and does not involve human subjects, data collection, or novel system deployment. Descriptions of compliance risks could in principle inform adversarial exploitation; we focus on detection and mitigation and omit implementation details. Industry data cited in §6 is self-reported by organizations with commercial interests. The healthcare and financial scenarios in §5 are constructed illustrations, not real incident reports.

References

Nirajan Acharya and Gaurav Kumar Gupta. 2026. A formal security framework for MCP-based AI agents: Threat taxonomy, verification models, and defense mechanisms. *arXiv preprint arXiv:2604.05969*.

Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, Zico Kolter, Matt Fredrikson, Yarin Gal, and Xander Davies. 2025. AgentHarm: A benchmark for measuring harmfulness of LLM agents. In *ICLR*.

Arkose Labs. 2026. 2026 agentic AI security report. Technical report, Arkose Labs. Survey of 300 enterprise leaders.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, and 1 others. 2022. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*.

Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. 2012. A survey of Monte Carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1):1–43.

Yuandao Cai, Wensheng Tang, Cheng Wen, and Shengchao Qin. 2026. Ghost in the agent: Redefining information flow tracking for LLM agents. *arXiv preprint arXiv:2604.23374*.

Mert Cemri, Melissa Z. Pan, Shuyi Yang, Lakshya A. Agrawal, Bhavya Chopra, Rishabh Tiwari, Kurt Keutzer, Aditya Parameswaran, Dan Klein, Kannan Ramchandran, Matei Zaharia, Joseph E. Gonzalez, and Ion Stoica. 2025. Why do multi-agent LLM systems fail? *arXiv preprint arXiv:2503.13657*. NeurIPS 2025.

Yen-Shan Chen, Sian-Yao Huang, Cheng-Lin Yang, and Yun-Nung Chen. 2026. Tracesafe: A systematic assessment of LLM guardrails on multi-step tool-calling trajectories. *arXiv preprint arXiv:2604.07223*.

Cloud Security Alliance and Token Security. 2026. Autonomous but not controlled: AI agent incidents now common in enterprises. Technical report, Cloud Security Alliance. Survey of 418 IT and security professionals. Published April 2026.

Manuel Costa, Boris Köpf, Aashish Kolluri, Andrew Paverd, Mark Russinovich, Ahmed Salem, Shruti Tople, and 1 others. 2025. Securing AI agents with information-flow control. *arXiv preprint arXiv:2505.23643*.

Edoardo Debenedetti, Ilia Shumailov, Tianqi Fan, Jamie Hayes, Nicholas Carlini, Daniel Fabian, Christoph Kern, Chongyang Shi, Andreas Terzis, and Florian Tramèr. 2025. Defeating prompt injections by design. *Preprint*, arXiv:2503.18813.

Edoardo Debenedetti, Jie Zhang, Mislav Balunovic, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. 2024. AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents.

Zehang Deng, Yongjian Guo, Changzhou Han, Wanlun Ma, Junwu Xiong, Sheng Wen, and Yang Xiang. 2025. AI agents under threat: A survey of key security challenges and future pathways. *ACM Computing Surveys*, 57(7):1–36.

Dorothy E. Denning. 1976. A lattice model of secure information flow. *Communications of the ACM*, 19(5):236–243.

Yi Dong, Ronghui Mu, Yanghao Zhang, Siqi Sun, Tianle Zhang, Changshun Wu, Gaojie Jin, Yi Qi, Jinwei Hu, Jie Meng, Saddek Bensalem, and Xiaowei Huang. 2025. Safeguarding large language models: A survey. *Artificial Intelligence Review*, 57(7):182:1–182:36.

European Parliament and Council. 2024. Regulation (EU) 2024/1689 of the european parliament and of the council (artificial intelligence act). *Official Journal of the European Union*.

Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, and 1 others. 2022. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*.

Gravitee. 2026. State of AI agent security 2026. Technical report, Gravitee. Survey of 919 executives and practitioners. Published February 2026.

Safayat Bin Hakim, Muhammad Adil, Alvaro Velasquez, Shouhuai Xu, and Houbing Herbert Song. 2026. Neuro-symbolic AI for cybersecurity: State of the art, challenges, and opportunities. *arXiv preprint arXiv:2509.06921*.

Gloria He and 1 others. 2026. Agent harness for large language model agents: A survey. *Preprints*. Published April 2026.

Xinyi Hou, Yanjie Zhao, Shuai Wang, and Haoyu Wang. 2025. Model context protocol (MCP): Landscape, security threats, and future research directions. *Preprint*, arXiv:2503.23278.

Wenyue Hua, Xianjun Yang, Mingyu Jin, Zelong Li, Wei Cheng, Ruixiang Tang, and Yongfeng Zhang. 2024. Trustagent: Towards safe and trustworthy LLM-based agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10000–10016.

IBM Research. 2025. Oneshield – the next generation of LLM guardrails. *arXiv preprint arXiv:2507.21170*.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. 2023. Llama guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*.

Gauri Kholkar and Ratinder Ahuja. 2025. The AI agent code of conduct: Automated guardrail policy-as-prompt synthesis. *arXiv preprint arXiv:2509.23994*.

Divyanshu Kumar, Nitin Aravind Birur, Tanay Baswa, Sahil Agarwal, and Prashanth Harshangi. 2025. No free lunch with guardrails. *arXiv preprint arXiv:2504.00441*.

Thomas Kuntz, Agatha Duzan, Hao Zhao, Francesco Croce, Zico Kolter, Nicolas Flammarion, and Maksym Andriushchenko. 2025. OS-Harm: A benchmark for measuring safety of computer use agents. *arXiv preprint arXiv:2506.14866*.

Leslie Lamport, Robert Shostak, and Marshall Pease. 1982. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401.

Frank Li. 2026. OpenClaw PRISM: A zero-fork, defense-in-depth runtime security layer for tool-augmented LLM agents. *arXiv preprint arXiv:2603.11853*.

Ninghui Li, Ziqing Mao, and Hong Chen. 2007. Usable mandatory integrity protection for operating systems. *IEEE Symposium on Security and Privacy*.

Ninghui Li, John C. Mitchell, and William H. Winsborough. 2002. Design of a role-based trust-management framework. In *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, pages 114–130. IEEE.

Ilija Lichkovski, Alexander Müller, Mariam Ibrahim, and Tiwai Mhundwa. 2025. EU-agent-bench: Measuring illegal behavior of LLM agents under EU law. *arXiv preprint arXiv:2510.21524*.

Zhenyu Mao, Jacky Keung, Fengji Zhang, Shuo Liu, Yifei Wang, and Jialong Li. 2025. Towards engineering multi-agent LLMs: A protocol-driven approach. *arXiv preprint arXiv:2510.12120*.

Kenneth L. McMillan. 1999. Circular compositional reasoning about liveness. In *Advances in Hardware Design and Verification (CHARME '99)*, volume 1703 of *Lecture Notes in Computer Science*, pages 342–345. Springer.

National Institute of Standards and Technology. 2023. AI Risk Management Framework (AI RMF 1.0). Technical Report NIST AI 100-1, NIST.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

OWASP GenAI Security Project. 2025. Owasp top 10 for agentic applications 2026. https://genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/. Released December 2025.

Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, and 1 others. 2023. Discovering language model behaviors with model-written evaluations. In *Findings of ACL*.

Charles Perrow. 1984. *Normal Accidents: Living with High-Risk Technologies*. Princeton University Press.

Traian Rebedea, Razvan Dinu, Makesh Narsimhan Sreedhar, Christopher Parisien, and Jonathan Cohen. 2023. Nemo guardrails: A toolkit for controllable and safe LLM applications with programmable rails. pages 431–445.

Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, Yann Dubois, Chris J. Maddison, and Tatsunori Hashimoto. 2023. Identifying the risks of LM agents with an LM-emulated sandbox. *arXiv preprint arXiv:2309.15817*.

Andrei Sabelfeld and Andrew C. Myers. 2003. Language-based information-flow security. *IEEE Journal on Selected Areas in Communications*, 21(1):5–19.

César Sánchez, Gerardo Schneider, Wolfgang Ahrendt, and 1 others. 2018. A survey of challenges for runtime verification from advanced application domains (beyond software). *arXiv preprint arXiv:1811.06740*.

Shoaib Ahmed Siddiqui, Radhika Gaonkar, Alex Aiken, Mihai Christodorescu, Arjun Guha, and Limin Jia. 2024. Permissive information-flow analysis for large language models. *arXiv preprint arXiv:2410.03055*.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, and 1 others. 2017. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359.

Merlin Stein. 2026. How are AI agents used? evidence from 177,000 MCP tools. *arXiv preprint arXiv:2603.23802*.

Simon Storf, Julius Berman, Leonid Proleev, Jason Hoelscher-Obermaier, Saffron Huang Jawhar, Sebastian Rose, and Owain Evans. 2026. Constitutional black-box monitoring for scheming in LLM agents. *arXiv preprint arXiv:2603.00829*.

Haoyu Wang, Christopher M. Poskitt, and Jun Sun. 2026a. Agentspec: Customizable runtime enforcement for safe and reliable LLM agents. In *ICSE '26*. ACM.

Haoyu Wang, Christopher M. Poskitt, Jiali Wei, and Jun Sun. 2026b. ProbGuard: Probabilistic runtime monitoring for LLM agent safety. *Preprint*, arXiv:2508.00500.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science*.

Peiran Wang, Yang Liu, Yunfei Lu, Yifeng Cai, Hongbo Chen, Qingyou Yang, Jie Zhang, Jue Hong, and Ye Wu. 2025a. AgentArmor: Enforcing program analysis on agent runtime trace to defend against prompt injection. *arXiv preprint arXiv:2508.01249*.

Xunguang Wang, Zhenlan Ji, Wenxuan Wang, Zongjie Li, Daoyuan Wu, and Shuai Wang. 2025b. Sok: Evaluating jailbreak guardrails for large language models. *arXiv preprint arXiv:2506.10597*.

Nell Watson, Isabelle Hupont, Songül Tolan, Hatice Gunes, and Iyad Rahwan. 2025. Personalized constitutionally-aligned agentic superego: Secure AI behavior aligned to diverse human values. *arXiv preprint arXiv:2506.13774*.

Jie Wu and Ming Gong. 2026a. Beyond single-agent alignment: Preventing context-fragmented violations in multi-agent systems. *arXiv preprint arXiv:2604.22879*.

Jie Wu and Ming Gong. 2026b. Policy-invisible violations in LLM-based agents. *arXiv preprint arXiv:2604.12177*.

Sheng Yin, Xianghe Pang, Yuanzhuo Ding, Menglan Chen, Yutong Bi, Yichen Xiong, Wenhao Huang, Zhen Xiang, Jing Shao, and Siheng Chen. 2024. Safeagentbench: A benchmark for safe task planning of embodied LLM agents. *arXiv preprint arXiv:2412.13178*.

Ethan G. Young and 1 others. 2019. The true cost of containing: A gVisor case study. In *HotCloud*.

Tongxin Yuan, Zhiwei He, Lingzhong Dong, Yiming Wang, Ruijie Zhao, Tian Xia, Lizhen Xu, Binglin Zhou, Fangqi Li, Zhuosheng Zhang, Rui Wang, and Gongshen Liu. 2024. R-judge: Benchmarking safety risk awareness for LLM agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1467–1490.

Yedi Zhang, Yi Emma Sun, Annabelle Jia En Lee, and Jin Song Dong. 2025. RvLLM: LLM runtime verification with domain knowledge. In *NeurIPS*.

Zhexin Zhang, Shiyao Cui, Yida Lu, Jingzhuo Zhou, Junxiao Yang, Hongning Wang, and Minlie Huang. 2024. Agent-safetybench: Evaluating the safety of LLM agents. *arXiv preprint arXiv:2412.14470*.

Yifan Zhou and 1 others. 2025. Mind the agent: A comprehensive survey on large language model-based agent safety. *UIUC Spring 2025 CS598 LLM Agent Workshop*.

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*.

A Supplementary Tables

B Compliance-Fragmentation Risks: Structural Intuition

This appendix provides a structured description of the compliance-fragmentation risk pattern discussed in the main text (§5). We describe the structural conditions under which such risks arise, rather than providing a full formal definition—the appropriate formalization depends on the deployment domain (temporal logic for process compliance, information-flow logic for data governance, etc.) and remains an open research question.

Structural conditions. Consider a multi-agent system $\mathcal{S} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$ executing a composite workflow. Compliance-fragmentation risks arise when three structural conditions co-occur:

1. **Local compliance:** Each agent $\mathcal{A}_i$ satisfies its own local policy for every action it takes, given the information available to it.
2. **Global constraint violation:** The composite workflow—considered as a whole and with full information—violates an organizational or regulatory constraint that spans multiple agents.
3. **Distributed information:** The evidence needed to detect the global violation is distributed across agent boundaries; no single agent possesses sufficient context to identify the violation independently.

Relationship to known concepts. This risk pattern is a specific instance of the well-studied *compositional verification problem* in distributed systems (McMillan, 1999): local correctness does not automatically compose into global correctness under partial observability. The novelty in the LLM-agent context lies in the additional complication of *semantic transformations*: agents communicate

Table 5: Master table of surveyed work. **H**: Enforcement Horizon—Act (Action), Traj (Trajectory), Fed (Federated). **M**: Enforcement Modality—Sym (Symbolic), IFC (Information-Flow), Emp (Empirical). See Figure 1. The Federated×IFC cell remains sparsely populated, suggesting an underdeveloped area rather than merely a table artifact.

Work	H	M	Technique	Threat	Topo.	Key Contribution
Pre-Execution Defenses (§3)						
NeMo Guardrails (Rebedea et al., 2023)	Act	Sym	Colang scripting	Both	S	Programmable conversational rails
Policy-as-Prompt (Kholkar and Ahuja, 2025)	Act	Emp	LLM-compiled classifiers	Non-Adv	S	Auto-translate docs to guardrails
Constitutional Mon. (Storf et al., 2026)	Act	Emp	Self-critique chains	Adv	S	Constitutional AI for agents
AgentSpec (Wang et al., 2026a)	Act	Sym	DSL predicates + enforcement	Non-Adv	S	Customizable runtime rules
RvLLM (Zhang et al., 2025)	Act	Sym	Propositional logic + FC	Non-Adv	S	Domain-expert ESL specs
Runtime Enforcement (§4)						
Fides (Costa et al., 2025)	Act	IFC	Dynamic taint tracking	Both	S	Formal IFC for LLM agents
Perm. IFA (Siddiqui et al., 2024)	Act–Traj	IFC	Influence-based label prop.	Both	S	Reduces over-tainting
TraceSafe (Chen et al., 2026)	Traj	Emp	Trajectory mutation bench.	Adv	S	Multi-step trace evaluation
NeuroTaint (Cai et al., 2026)	Traj	IFC	Semantic taint + causal probes	Adv	S	Outperforms Fides on TaintBench
AgentArmor (Wang et al., 2025a)	Traj	IFC	PDG-based trace analysis	Adv	S	ASR $<$3%, 1% utility loss
MCP-Guard† (Acharya and Gupta, 2026)	Act	Sym	Tool-call interception	Adv	S	MCP-layer guardrails
DeceptGuard (Debenedetti et al., 2025)	Act	Sym	Deceptive tool detection	Adv	S	Tool description poisoning
OneShield (IBM Research, 2025)	Act	Emp	Multi-detector pipeline	Both	S	Classification + extraction
No Free Lunch (Kumar et al., 2025)	Act	Emp	Empirical tradeoff analysis	Both	S	Safety–utility–usability tradeoff
Wu and Gong (2026b)	Traj	IFC	Graph invariant monitoring	Non-Adv	S	Policy-invisible violation detect.
Multi-Agent Solutions (§5)						
MAST (Cemri et al., 2025)	Fed	Emp	Failure taxonomy + dataset	Both	M	14 failure modes, 1600+ traces
SEMAP (Mao et al., 2025)	Fed	Sym	Protocol-layer contracts	Non-Adv	M	Behavioral contracts for MAS
MCP Security (Hou et al., 2025)	Fed	Sym	Threat taxonomy (16 scen.)	Adv	M	MCP lifecycle threat model
A2A Threat Mod.† (Acharya and Gupta, 2026)	Fed	Sym	Cross-protocol analysis	Adv	M	MCP/A2A/ACP/ANP comparison
Wu and Gong (2026a)	Fed	IFC	Distributed taint tracking	Non-Adv	F	Cross-domain compliance detection
Benchmarks & Evaluation (§6)						
AgentHarm (Andriushchenko et al., 2025)	Act	Emp	440 malicious tasks	Adv	S	Misuse robustness benchmark
Agent-SafetyBench (Zhang et al., 2024)	Act	Emp	2000 test cases, 8 risk cat.	Both	S	Comprehensive safety eval
ToolEmu (Ruan et al., 2023)	Traj	Emp	LLM-emulated sandbox	Adv	S	Scalable agent risk testing
EU-Agent-Bench (Lichkovski et al., 2025)	Act	Sym	EU law compliance	Non-Adv	S	Legal compliance benchmark
R-Judge (Yuan et al., 2024)	Traj	Emp	Safety risk recognition	Both	S	Trace-level safety judgment
OS-Harm (Kuntz et al., 2025)	Act	Emp	Computer-use agent safety	Adv	S	GUI agent safety benchmark
TrustAgent (Hua et al., 2024)	Traj	Sym	Safe & trustworthy planning	Both	S	Agent Constitution framework
Surveys & Systematizations (§1)						
Mind the Agent (Zhou et al., 2025)	–	–	4-threat taxonomy	Adv	S/M	Agent safety survey
SoK Guardrails (Wang et al., 2025b)	–	–	6-dim. taxonomy + SEU	Adv	S	Guardrail eval framework
Dong et al. (Dong et al., 2025)	–	–	Guardrail landscape	Both	S	LLM safeguarding survey
Agent Harness (He et al., 2026)	–	–	LTS semantics	Both	S/M	Harness engineering survey
Deng et al. (Deng et al., 2025)	–	–	ACM CSUR agent threats	Adv	S/M	Comprehensive threat survey
Additional notable systems						
PRISM (Li, 2026)	Traj	Sym	Defense-in-depth runtime	Both	S	Zero-fork agent security layer
Superego (Watson et al., 2025)	Act	Emp	Constitutional oversight	Both	S	Value-aligned oversight

†MCP-Guard and A2A Threat Mod. are two aspects of the same security framework (Acharya and Gupta, 2026), listed separately because they operate at different horizons.

through natural-language summaries that may preserve task-relevant content while losing policy-relevant metadata (the "semantic laundering" phenomenon discussed in §4.2).

Current empirical status. To our knowledge, no public incident report has documented a confirmed compliance-fragmentation event in deployed multi-agent LLM systems. The pattern remains a prospective risk model, motivated by the architectural properties of current MAS designs (partial observability, heterogeneous policies, provenance loss at handoffs). We include it in this survey because the *structural preconditions* are clearly present in production systems, and because existing benchmarks do not evaluate this risk category (§6).

C Mapping to Regulatory Frameworks

Table 8 provides an illustrative mapping between enforcement mechanisms and governance concerns discussed in three major AI governance frameworks. This mapping is intended to help practitioners reason about which technical approaches may be relevant to specific governance obligations.

D Detection of Compositional Compliance Risks: Open Challenges

Detecting compositional compliance risks in multi-agent workflows remains an open research chal-

Table 6: Comparison of production guardrail tools and frameworks. **Deploy**: Cloud (hosted API), Self (self-hosted). **Enf.**: Input (I), Output (O), Bidirectional (B), Action-level (A). **Agent**: agentic workflow support. **MCP**: Model Context Protocol support. **Cust.**: custom policy support. **OSS**: open-source availability. ✓= yes, ○= partial, ✗= no.

Tool / Framework	Deploy	Enf.	Agent	MCP	Cust.	OSS
NVIDIA NeMo Guardrails	Self	B	○	✗	✓	✓
Guardrails AI	Self	O	○	✗	✓	✓
LlamaGuard (Meta)	Self	I/O	✗	✗	○	✓
AWS Bedrock Guardrails	Cloud	B	○	✗	✓	✗
Azure AI Content Safety	Cloud	B	○	✗	✓	✗
Cisco AI Defense	Cloud	A	✓	✓	✓	✗
Prompt Guard (Meta)	Self	I	✗	✗	✗	✓
WildGuard	Self	I/O	✗	✗	✗	✓
Bifrost (Maxim AI)	Self	B	✓	✗	✓	✓
Harmonic MCP Gateway	Cloud	A	✓	✓	✓	✗
AccuKnox (CNAPP)	Cloud	A	✓	○	✓	○
Arthur AI Shield	Cloud	B	✓	○	✓	✗

Note: Lakera Guard (acquired by Cisco, May 2025) is subsumed under Cisco AI Defense.

Table 7: Summary of large-scale industry surveys on AI agent security (2026). All figures are self-reported. **N**: sample size. **Population**: who was surveyed. **Method**: data collection approach. Key findings are grouped by theme for cross-survey comparison.

Survey	N	Date	Method	Key Findings
CSA (Cloud Security Alliance and Token Security, 2026)	418	Jan 2026	Online survey (IT/security)	• 82% discovered unknown (shadow) agents • 65% had agent-related incidents
Gravitee (Gravitee, 2026)	919	Feb 2026	Online survey (exec. + eng.)	• 88% reported security incidents • Only 14.4% with full security approval
Arkose Labs (Arkose Labs, 2026)	300	Q1 2026	Survey (Fortune 100-scale)	• 97% expect incident within 12 months • Only 6% of budgets for agentic AI security

lenge. We briefly outline the key technical requirements rather than proposing a specific algorithm, as the design space is still poorly understood.

A viable detection approach would need to address at least three subproblems: (1) *provenance tracking*—recording which upstream agent outputs causally influenced each downstream action, ideally through lightweight metadata propagation at agent handoff points; (2) *global policy evaluation*—checking the assembled provenance graph against cross-agent compliance predicates that cannot be decomposed into independent local checks; and (3) *semantic comparison*—determining whether LLM-mediated reformulations preserve policy-relevant content, which requires domain-specific similarity thresholds that remain difficult to calibrate.

Early explorations of structural taint tracking suggest that counterfactual simulation—testing whether removing a tainted input changes the downstream output—is a promising direction. Key research challenges include managing inference cost across inter-agent handoffs, developing standardized benchmarks for detection accuracy, and distinguishing benign summarization from policy-relevant information loss. We consider the design of efficient, accurate compositional compliance detectors to be among the most important open problems in multi-agent safety (§7).

E Extended Benchmark Comparison

Table 9 extends the benchmark comparison in §6 with additional dimensions: code availability, license type, and whether the benchmark supports automated evaluation without human annotation.

Table 8: Illustrative mapping of action-level safety techniques to governance concerns. ✓ = directly relevant, ○ = partially relevant, ✗ = not a primary fit. This table is not a legal compliance assessment. EU AI Act obligations effective August 2026 (European Parliament and Council, 2024); NIST AI RMF (National Institute of Standards and Technology, 2023); OWASP Agentic Top 10 (OWASP GenAI Security Project, 2025).

Technical Mechanism	**HM Cell**	**EU AI Act**	**NIST RMF**	**OWASP Agentic**
Symbolic / Rule-Based				
Policy specification (AgentSpec, RvLLM)	Act × Sym	✓	✓	✓
Capability bounding / schema filtering	Act × Sym	✓	✓	✓
Temporal logic monitoring	Traj × Sym	○	✓	○
Federated policy protocols (SEMAP, MCP)	Fed × Sym	○	○	✓
Information-Flow / Taint				
Dynamic taint tracking (Fides)	Act/Traj × IFC	✓	✓	✓
Permissive IFA (over-taint reduction)	Act/Traj × IFC	○	✓	○
Cross-agent taint (STT Protocol)	Fed × IFC	✓	○	✓
Semantic laundering detection	Traj/Fed × IFC	○	○	○
Empirical / Learned				
LLM-as-judge monitoring	Act/Traj × Emp	○	✓	○
Adversarial robustness testing	Act × Emp	✗	✓	✓
Multi-agent simulation / MAST	Fed × Emp	✗	○	○
Behavioral anomaly detection	Traj/Fed × Emp	○	✓	✓
Underdeveloped governance concern				
Non-adversarial compliance detection	Fed × all	✓*	○	✗
Standardized compliance metrics	cross-cutting	✓*	○	✗

*Related to EU AI Act Art. 9 (risk management) and Art. 13 (transparency), though no technical standard yet exists.

Table 9: Extended agent safety benchmark comparison. **Auto**: fully automated evaluation (no human annotation required). **Code**: publicly available code. **License**: dataset/code license. **Multi-Ag**: multi-agent scenario support. ✓=yes, ○=partial, ✗=no.

Benchmark	**Venue**	**Scale**	**Auto**	**Code**	**License**	**Multi-Ag**	**Non-Adv**	**HM Cell**
AgentHarm (Andriushchenko et al., 2025)	ICLR'25	440	✓	✓	MIT	✗	✗	Act/Emp
Agent-SafetyBench (Zhang et al., 2024)	arXiv'24	2,000	✓	✓	Apache-2	✗	○	Act/Emp
ToolEmu (Ruan et al., 2023)	ICLR'24	144	○	✓	MIT	✗	✓	Act/Emp
TraceSafe (Chen et al., 2026)	arXiv'26	900+	✓	✗	—	✗	✗	Traj/Emp
EU-Agent-Bench (Lichkovski et al., 2025)	arXiv'25	200+	✓	✗	—	✗	✓	Act/Sym
OS-Harm (Kuntz et al., 2025)	NeurIPS'25	300+	○	✓	MIT	✗	✗	Traj/Emp
SafeAgentBench (Yin et al., 2024)	arXiv'24	750	✓	✓	Apache-2	○	✓	Act/Emp
R-Judge (Yuan et al., 2024)	EMNLP'24	162	✓	✓	CC-BY	✗	○	Traj/Emp
MAST-Data (Cemri et al., 2025)	NeurIPS'25	1,600+	✓	✓	Apache-2	✓	✓	Fed/Emp
Agent-SafetyBench (v2)	arXiv'25	4,000+	✓	○	—	○	○	Act/Emp
AgentDojo (Debenedetti et al., 2024)	NeurIPS'24	97	✓	✓	MIT	✗	○	Act/Emp
Needed (compliance bench.)	—	—	✓	✓	Open	✓	✓	*Fed/all*