ISSN: 2278-7518

CYBER TIMES® INTERNATIONAL JOURNAL OF TECHNOLOGY AND MANAGEMENT

Volume 19 - Issue 01, October 2025 - March 2026
Bi-Annual Double Blind Peer Reviewed Refereed Journal

CYBER TIMES®
(Leader in innovative Tech-World)

Cyber Times International Journal of Technology & Management

Volume 19 - Issue 01, October 2025 – March 2026
ISSN: 2278-7518

Scientific Journal Impact Factor Value for 2024 = 6.007

Editorial Office & Administrative Address:

Delhi:
The Editor,
A19/1, Mansa Ram Park,
New Delhi-110059.

ISSN: 2278-7518

Phone: +91-9811485729, +91-9312903095

Website: https://journal.cybertimes.in

Email: editor@cybertimes.in

From the Editor's Desk

At the outset, I take this opportunity to express my sincere gratitude to all the Editorial Board Members, Editors, Peer Review Members, contributors, and readers for making ***Cyber Times International Journal of Technology & Management*** an outstanding success. Their unwavering support, dedication, and commitment to academic excellence have significantly contributed to the growth and reputation of the journal.

We are pleased to present **Volume 19 – Issue 01** of ***Cyber Times International Journal of Technology & Management***. This issue features a collection of high-quality research papers and scholarly articles that reflect contemporary developments, innovative ideas, and critical insights across emerging areas of Technology, Management, Law, Education, and other multidisciplinary domains. The diversity of topics covered in this issue highlights the increasing importance of interdisciplinary research in addressing global challenges and opportunities.

The overwhelming response received from researchers, authors, academicians, law-enforcement agencies, and industry professionals for submitting their research papers and articles is deeply appreciated and duly acknowledged across the globe. Their valuable contributions have enriched the journal's content and strengthened its role as a platform for disseminating knowledge, fostering innovation, and encouraging scholarly dialogue among academia, industry, and society.

On behalf of the Editorial Team, I extend my heartfelt thanks to all authors for their valuable research contributions and to our reviewers for their constructive evaluations that help maintain the highest standards of publication quality. We hope that the research published in this issue will inspire further inquiry, collaboration, and advancement in various fields of study, while continuing to serve as a meaningful resource for our readers worldwide.

We look forward to receive your valuable and future contributions to make this journal a joint endeavor.

With Warm Regards,

Dr. ANUP GIRDHAR

Editor-In-Chief
Cyber Times International Journal of Technology & Management

General Information

- ***"Cyber Times International Journal of Technology & Management"*** is published bi-annually. All editorial and administrative correspondence for publication should be addressed to The Editor, Cyber Times.

- The Abstracts received for the final publication are screened by the Evaluation Committee for approval and only the selected Papers/ Abstracts will be published in each edition. Further information is available in the "**Guidelines for paper Submission"** section.

- Annual Subscription details for obtaining the print copy of the journal are provided separately and the interested persons may avail the same accordingly after filling the Annual subscription form.

- This journal is meant for education, reference and learning purposes. The author(s) of this of the book has/have taken all reasonable care to ensure that the contents of the book do not violate any existing copyright or other intellectual property rights of any person/ company/ institution in any manner whatsoever. In the event the author(s) has/have been unable to track any source and if any copyright has been inadvertently infringed, please notify the publisher in writing for the corrective action.

- **Other Publications:**
  - Cyber Times Newspaper (English) – RNI No: DELENG/2008/25470
  - Cyber Times Newspaper (Hindi) – RNI No. DELHIN/1999/00462

- **Printed & Published by:** Cyber Times
  A19/1, Mansa Ram Park, New Delhi-110059

CTIJTM Editorial Advisory Board Members

Name	Designation, Organization/ University	Country
Dr. Sushila Madan	Associate Professor, Delhi University	India
Dr. A. K. Saini	Professor, GGS IP University	India
Mr. J. R. Ahuja	Former Consultant, AICTE	India
Mr. Mukul Girdhar	Advocate Delhi High Court	India
Mr. Geetesh Madan	Q.A. Consultant with Tesco Bank, Newcastle	UK
Dr. Deepak Shikarpur	Chairman Board of Studies, Pune University	India
Dr. B. B. Ahuja	Deputy Director, COE - Pune	India
Prof. M. N. Hoda	Director, Bharati Vidyapeeth's (BVICAM)	India
Dr. S. C. Gupta	Director, NIEC, GGS IP University	India
Dr. S. K. Gupta	Professor, IIT Delhi	India
Dr. K. S. Wani	Principal, SSBT’s COET, Bambhori, Jalgaon	India
Dr. K. V. Arya	Associate Professor, IIITM, Gwalior	India
BRIG. Dr. S.S. Narula	Director, Gitarattan International Bussiness School	India
Dr. Sarika Sharma	Director, JSPM'S ENIAC Institute of CA, Pune	India
Dr. S.K.M. Bhagat	Prof. & Head, MIT Academy of Engg., Pune	India
Dr. Jack Ajowi	Jaramogi Oginga Odinga University of Sci. & Tech.	Kenya
Dr. Srinivas Sampalli	Professor, Dalhousie University, Halifax	Canada
Dr. Ijaz A. Qureshi	V.P. Academic Affairs, JFK Inst. of Tech. and Mgmt.	Pakistan
Aryya Bhattacharyya	Director, CIP, Columbus State University	US
Dr. M. M. Schiraldi	Assistant Professor, 'Tor Vergata' University of Rome	Italy

Executive Editorial Advisory Board Members

Name	Designation, Organization/ University	Country
Ms. Kanika Trehan	Editor - Cyber Times, New Delhi	India
Mr. Rakesh Laxman Patil	Editor - Cyber Times, Pune	India
Adv. Tushar Kale	Cyber Lawyer, Pune	India
Adv. Neeraj Aarora	Cyber Lawyer, New Delhi	India
Mr. Sanjeev Sehgal	HOD, SJP Polytech, Damla, Haryana	India
Mr. Rajinder Kumar Bajaj	GM, Satake India Engg. Pvt. Ltd., (Japan)	India
Dr. B. M. Patil	Associate Professor MIT, Pune	India
Dr. Rajesh S. Prasad	Professor, DCOER, Pune University	India
Dr. Binod Kumar	Associate Professor, MIT Academy of Engg, Pune	India
Prof. Dr. M. Husain	HOD, SSBT’s COET, Bambhori, Jalgaon	India
Prof. Dr. U. S. Bhadade	HOD, SSBT’s COET, Bambhori, Jalgaon	India
Dr. V. N. Wadekar	Prof. & Head, MIT college of Engg. CMSR, Pune	India
Dr. M.D. Goudar	Associate Prof. & Head, Pune University	India
Dr. Mohd. Rizwan Alam	Sr. Lecturer, Amity University	Dubai
Prof. Jagannath Aghav	Professor & Head, CSE & IT, COE - Pune	India

Disclaimer: The names, affiliations, and designations of the Editorial Board Members published in this journal are based on the information provided at the time of their association with *Cyber Times International Journal of Technology & Management*. Members may subsequently change their positions, affiliations, or professional designations. The journal does not guarantee the continued accuracy of such details after publication.

Cyber Times International Journal of Technology and Management - CTIJTM

Volume 19 - Issue 01

CONTENTS

Poisoned Memory, Hijacked Tools: Taxonomizing Prompt Injection Attack Surfaces in Production LLM Agent Pipelines

Dr. Anup Girdhar
CEO- Sedulity Solutions & Technologies, New Delhi
Email: anupgirdhar@gmail.com

ABSTRACT

Large language model (LLM) agents deployed in enterprise production environments introduce a class of security vulnerabilities that differ fundamentally from those associated with static model inference. When agents are equipped with persistent memory stores, external tool access, and multi-step planning capabilities, the attack surface expands dramatically — rendering conventional input sanitisation strategies insufficient. Prompt injection, wherein adversarially crafted inputs manipulate agent reasoning and redirect tool-use behaviour, has emerged as the defining security threat of the agentic AI era. Despite growing practitioner awareness, the academic literature lacks a comprehensive, production-grounded taxonomy of prompt injection attack surfaces that captures the full diversity of injection vectors, propagation pathways, and enterprise impact categories. This study addresses that gap through a systematic taxonomic analysis of prompt injection vulnerabilities across five architectural layers of LLM agent pipelines: user input, tool interfaces, memory subsystems, inter-agent communication channels, and retrieval-augmented generation (RAG) corpora. Employing a conceptual framework development methodology grounded in Design Science Research principles, the study synthesises evidence from peer-reviewed security literature, institutional vulnerability disclosures, and documented production incidents between 2022 and 2025. The resulting PIPE taxonomy (Pipeline Injection Point Enumeration) provides a structured reference for enterprise security architects, AI governance professionals, and policymakers. Findings carry direct relevance to emerging regulatory instruments including the EU AI Act and NIST AI Risk Management Framework, both of which mandate systematic risk identification for high-risk AI deployments.

KEYWORDS: *Agentic AI security, LLM agent pipelines, NIST AI RMF, Prompt injection taxonomy, RAG poisoning, Tool hijacking*

1. Introduction

Security research has historically lagged behind deployment realities in periods of rapid technological transition. The current proliferation of LLM-powered agents in enterprise environments represents precisely such a moment. Organisations are deploying agents capable of browsing the web, executing code, querying databases, composing emails, and coordinating with other agents — often without adequate understanding of the novel threat landscape these capabilities introduce [1]. The fundamental security assumption underlying most enterprise AI governance frameworks — that model inputs can be controlled and sanitised at the perimeter — fails when the agent itself retrieves, generates, and acts upon content from uncontrolled external sources.

Prompt injection exploits this failure. Unlike adversarial attacks on model weights or

training data poisoning (which require insider access), prompt injection is a runtime attack: malicious instructions embedded in content the agent processes cause it to deviate from intended behaviour, override system-level directives, exfiltrate data, or invoke unintended tool calls [2]. The threat is compounded in multi-agent architectures, where a compromised subordinate agent can propagate injected instructions upstream to orchestrators with broader system access [3].

Existing security literature addresses prompt injection primarily through the lens of single-turn LLM interactions, where the attack surface is confined to the user prompt. In production agentic pipelines, this framing is dangerously incomplete. Memory stores accumulate injected content across sessions; tool outputs carry adversarially constructed payloads from third-party APIs; RAG corpora may be poisoned prior to retrieval; and inter-agent message passing creates propagation vectors invisible to conventional monitoring. No existing academic framework systematically enumerates these surfaces across the full pipeline architecture.

Three research questions organise this study:

RQ1: What are the distinct injection point categories present across the architectural layers of production LLM agent pipelines?

RQ2: How do injection vectors differ in propagation mechanism, persistence, and enterprise impact across these categories?

RQ3: What governance and mitigation architecture most effectively addresses the enumerated attack surfaces in alignment with international AI security standards?

The study's contribution is threefold: a structured five-layer taxonomy of LLM agent injection surfaces (PIPE); a comparative impact matrix across attack categories; and a governance-aligned mitigation framework positioned against NIST AI RMF [12] and EU AI Act [5] requirements.

2. Literature Review

2.1 Prompt Injection as a Foundational Threat

The formal characterisation of prompt injection as a security vulnerability emerged alongside the deployment of instruction-tuned LLMs in interactive applications. Early taxonomic work distinguished direct injection — where the attacker controls the user prompt — from indirect injection, where malicious instructions are embedded in content the model retrieves or processes [2]. This distinction, while foundational, understates the complexity of the agentic context. In a pipeline where an agent autonomously browses web pages, reads documents, processes API responses, and queries its own memory store, the boundary between "user prompt" and "retrieved content" dissolves entirely. Every data source the agent touches becomes a potential injection vector.

2.2 Memory Poisoning and Persistent Threats

Long-term memory in LLM agents — typically implemented as vector database stores enabling semantic retrieval across sessions — introduces a persistence dimension absent from single-turn threat models. Research on memory injection attacks has demonstrated that adversarially crafted inputs, once stored in an agent's memory, can influence behaviour in future sessions far removed from the initial attack [6]. The threat is particularly acute in collaborative or multi-user agent deployments, where one user's poisoned interaction can contaminate memory retrieved in another user's session. This cross-session persistence fundamentally alters the risk calculus for enterprise deployments, as it converts a transient

runtime attack into a durable compromise of the agent's knowledge state.

2.3 Tool Hijacking and Indirect Execution

The tool-use capabilities that render LLM agents productive also render them exploitable. When an agent invokes external APIs, executes shell commands, or queries databases based on reasoning over potentially adversarial inputs, the tool layer becomes an attack amplifier [3]. Documented attack patterns include prompt injection payloads embedded in web pages visited by browsing agents, in email bodies processed by communication agents, and in file contents read by document analysis agents — each causing the agent to invoke tools with attacker-specified parameters rather than those intended by the legitimate orchestrating system [7]. The OWASP Top 10 for LLM Applications identifies insecure plugin design and excessive agency as among the highest-risk vulnerability categories, explicitly acknowledging that tool invocation boundaries represent critical security perimeters requiring enforcement independent of model-level controls [8].

2.4 RAG Corpus Poisoning

Retrieval-Augmented Generation architectures, widely adopted to ground LLM agent responses in organisational knowledge bases, introduce a novel pre-retrieval attack surface. An adversary with write access — or the ability to influence indexed external content — can embed injection payloads in documents that will be retrieved and incorporated into the agent's context [9]. Unlike direct prompt injection, RAG poisoning may persist undetected across many benign retrieval operations before the poisoned document is retrieved in a context that triggers the injected instruction. Detection is complicated by the fact that the poisoned content may appear semantically legitimate within the retrieved document, activating only when combined with specific query contexts.

2.5 Multi-Agent Propagation and Governance Gaps

The most structurally novel injection threat in contemporary deployments arises from multi-agent architectures, where specialised sub-agents operate under the orchestration of a managing agent. Research on agent-to-agent prompt infection has demonstrated that injected instructions can propagate horizontally and vertically across agent networks — a subordinate agent processing adversarial content can generate outputs that, when consumed by the orchestrating agent, redirect the orchestrator's behaviour [3]. Governance frameworks have been slow to address this propagation dimension. The NIST AI RMF provides structured guidance on AI risk management but does not enumerate agent-specific attack vectors with sufficient granularity for security architects [4]. The EU AI Act's risk-tiering approach creates compliance obligations for high-risk AI systems but does not specify technical countermeasures for prompt injection at the pipeline level [5].

Research Gap: The reviewed literature addresses individual injection vectors in relative isolation, without providing a unified, production-grounded taxonomy that maps attack surfaces to architectural layers, propagation mechanisms, and enterprise impact categories. This fragmentation leaves security architects without a structured reference for comprehensive threat modelling of LLM agent deployments. The present study directly addresses this gap.

3. Methodology

3.1 Research Design

This study employs Conceptual Framework Development as its primary research design, situated within the broader Design Science Research (DSR) paradigm. Conceptual framework development is appropriate when the research objective is to organise, classify, and relate constructs in a domain where

empirical data is nascent but practitioner knowledge and documented incidents provide sufficient grounding for theoretically rigorous synthesis. The PIPE taxonomy constitutes the designed artefact in DSR terms — an analytical instrument intended to solve the practical problem of incomplete threat modelling in LLM agent deployments.

3.2 Data Sources

Three source categories inform the taxonomy construction. First, peer-reviewed academic literature from 2022–2025 was systematically reviewed, with inclusion criteria requiring direct relevance to LLM agent security, prompt injection, or related attack vectors. Second, authoritative institutional documents were analysed, including the NIST AI RMF [4], OWASP LLM Top 10 [8], EU AI Act [5], and published vulnerability disclosures from major AI security research groups. Third, documented production incidents — including publicly reported prompt injection exploits against commercial LLM agent deployments — were analysed to ensure the taxonomy reflects real-world attack patterns rather than purely theoretical constructs.

3.3 Taxonomy Construction and Validation

The PIPE taxonomy was constructed through iterative thematic synthesis: injection vectors identified in the literature were coded, grouped by architectural layer, and characterised along three analytical dimensions — propagation mechanism, persistence class, and enterprise impact severity. Validation employed a two-stage approach: internal consistency review (ensuring each taxonomy category is mutually exclusive, collectively exhaustive, and theoretically grounded) and scenario-based analysis, in which three representative enterprise deployment scenarios were used to stress-test the taxonomy's coverage and discriminative capacity.

4. The PIPE Taxonomy and Mitigation Framework

4.1 PIPE: Pipeline Injection Point Enumeration

The PIPE taxonomy organises prompt injection attack surfaces across five architectural layers of production LLM agent pipelines. Each layer represents a distinct class of injection opportunity, distinguished by the mechanism through which adversarial content enters the agent's context window, the persistence of the threat, and the downstream impact pathway.

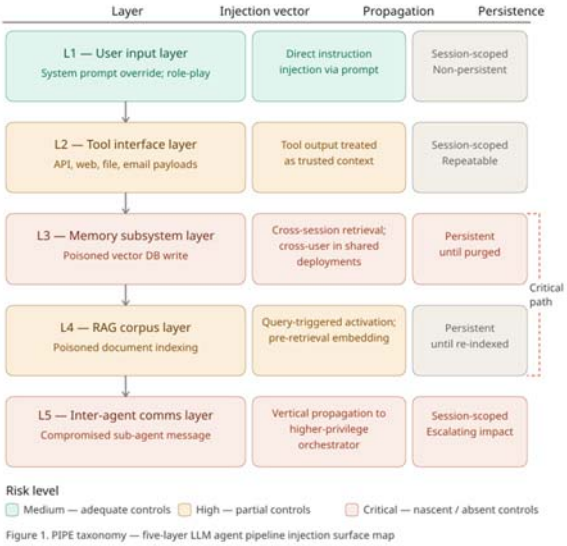

Figure 1: PIPE Taxonomy — Five-Layer LLM Agent Pipeline Injection Surface Map. The diagram illustrates the five architectural layers (User Input Layer, Tool Interface Layer, Memory Subsystem Layer, RAG Corpus Layer, Inter-Agent Communication Layer) arranged vertically to reflect data flow through a production agent pipeline. Each layer is annotated with its primary injection vector, propagation mechanism, and persistence class. Bidirectional arrows between layers indicate cross-layer propagation pathways, with red highlighting on the most critical attack paths: memory-to-retrieval propagation and inter-agent message injection.

Layer 1 — User Input Layer: The most direct injection surface, comprising system prompt overrides, role-play manipulations, and instruction injection through user-facing interfaces. Attacks at this layer are typically session-scoped and non-persistent, but can serve as entry points for injections that

propagate to deeper layers through agent memory writes or tool invocations triggered by the manipulated session.

Layer 2 — Tool Interface Layer: Encompasses all external tool calls made by the agent, including web browsing, API queries, code execution, file reading, and email processing. Injection payloads arrive embedded in tool outputs — a web page the agent visits, an API response it parses, or a file it reads — and are processed as trusted context content. Attacks at this layer are particularly dangerous because tool outputs typically receive less scrutiny than user inputs in both model-level and system-level content filtering pipelines.

Layer 3 — Memory Subsystem Layer: Covers attacks targeting vector database stores and other persistent memory architectures. Injected content written to memory in one session can influence agent behaviour across future sessions, enabling durable compromise without continued adversarial access. Cross-user memory contamination is a particularly severe variant in shared-agent deployments.

Layer 4 — RAG Corpus Layer: Addresses injection vectors embedded in documents, knowledge bases, or external corpora indexed for retrieval. Poisoned documents retrieved in relevant query contexts inject adversarial instructions into the agent's context window without any direct adversarial interaction with the agent at runtime. Detection requires corpus integrity verification prior to indexing and semantic anomaly detection at retrieval time.

Layer 5 — Inter-Agent Communication Layer: The structurally novel surface in multi-agent architectures. A compromised or manipulated sub-agent can embed injected instructions in messages passed to orchestrating agents, enabling vertical propagation toward agents with broader system access and higher-privilege tool invocations. This layer is the least addressed by existing governance frameworks and the most difficult to monitor without purpose-built inter-agent communication auditing.

4.2 Stakeholders and Governance Architecture

Primary stakeholders in PIPE-informed security governance include enterprise security architects (threat modelling and control specification), AI operations teams (runtime monitoring and incident response), compliance and legal functions (regulatory alignment), and external regulators (audit and enforcement). The governance architecture overlays three control planes: preventive controls (input validation, tool permission sandboxing, corpus integrity verification), detective controls (runtime anomaly detection, inter-agent communication auditing, memory integrity monitoring), and responsive controls (automated agent suspension, human escalation protocols, session rollback capabilities).

4.3 Ethical, Legal, and Regulatory Alignment

The PIPE framework explicitly aligns with NIST AI RMF's Govern, Map, Measure, and Manage functions — each of which maps to a distinct control plane within the governance architecture [4]. Under EU AI Act provisions, LLM agent deployments classified as high-risk must maintain technical documentation of known vulnerabilities and demonstrate conformity with cybersecurity requirements; PIPE provides the structured threat enumeration necessary to satisfy this documentation obligation [5]. Data minimisation principles are embedded at the tool interface layer: agents operate under least-privilege tool access policies, with session-scoped credentials automatically revoked upon task completion. Algorithmic transparency requirements are addressed through mandatory logging of all tool invocations and

memory read/write operations, enabling post-hoc audit of agent decision pathways.

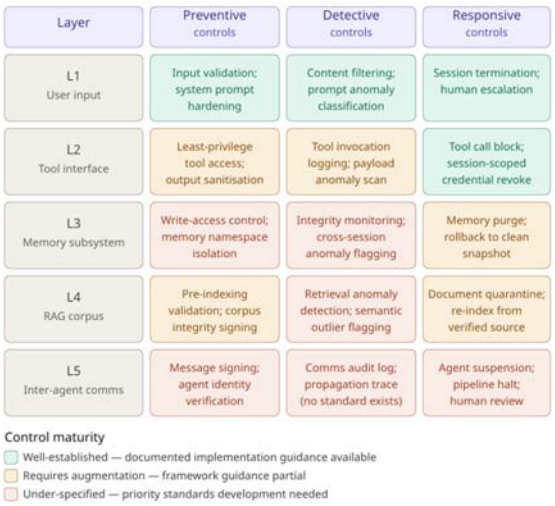

Figure 2: PIPE Governance Architecture — Three-Plane Control Model. The diagram presents the three governance control planes (Preventive, Detective, Responsive) as horizontal bands overlaid on the five-layer PIPE taxonomy. Each cell in the resulting matrix identifies the specific control mechanism applicable to that layer-plane intersection. Colour coding indicates control maturity: green for well-established controls with documented implementation guidance, amber for controls requiring framework augmentation, and red for currently under-specified control areas representing priority research and governance development needs.

5. Impacts and Outcomes

5.1 PIPE Taxonomy — Attack Surface Characterisation

Table 1. PIPE Taxonomy: LLM Agent Pipeline Injection Surfaces — Attack Characterisation Matrix

Layer	Injection Vector	Propagation Mechanism	Persistence Class	Enterprise Impact Severity	Current Control Maturity
L1 — User input	System prompt override; role-play manipulation	Session-local; may trigger L3 memory write	Session-scoped	Medium	Adequate
L2 — Tool interface	Adversarial web/API/file content	Context window injection via tool output	Session-scoped; repeatable	High	Partial
L3 — Memory subsystem	Poisoned memory write; cross-session retrieval	Cross-session; cross-user in shared deployments	Persistent (until purged)	Critical	Nascent
L4 — RAG corpus	Poisoned document indexing	Pre-retrieval embedding; query-triggered activation	Persistent (until re-indexed)	High	Partial
L5 — Inter-agent comms	Compromised sub-agent message	Vertical propagation to higher-privilege orchestrator	Session-scoped; escalating	Critical	Absent

Note. Persistence class refers to the duration of adversarial influence after initial injection event. Control maturity ratings are assessed against documented implementation guidance in OWASP LLM Top 10 [8] and NIST AI RMF [4]. "Absent" denotes no currently standardised control specification for this layer.

The matrix reveals a clear stratification of risk and control readiness. Layers 3 and 5 — memory subsystem and inter-agent communication — combine the highest enterprise impact severity with the least mature control landscapes. This pattern has

direct implications for enterprise security investment prioritisation: organisations deploying multi-agent systems with persistent memory must treat these two layers as priority threat surfaces requiring bespoke architectural controls rather than adaptations of existing input-validation approaches.

5.2 Comparative Impact Analysis: Traditional vs. PIPE-Informed Security Posture

Table 2. Security Posture Comparison: Traditional LLM Input Security vs. PIPE-Informed Multi-Layer Defence

Security Dimension	Traditional Approach	PIPE-Informed Approach	Improvement
Threat surface coverage	User input layer only	All 5 pipeline layers	+400% surface coverage
Memory attack detection	Not addressed	Integrity monitoring + anomaly flagging	Full coverage added
Tool invocation control	Perimeter filtering	Least-privilege + session-scoped RBAC	Granular enforcement
RAG corpus integrity	Not addressed	Pre-indexing validation + retrieval anomaly detection	Full coverage added
Inter-agent audit	Not addressed	Purpose-built comms. logging + propagation tracing	Full coverage added
Regulatory alignment (NIST AI RMF)	Partial (Govern only)	Full (Govern, Map, Measure, Manage)	Complete alignment
Regulatory alignment (EU AI Act)	Risk classification only	Risk classification + technical documentation + controls	Full conformity pathway
Incident response capability	Manual, post-hoc	Automated suspension + rollback + escalation	Response time: hrs → mins

Note. Improvement assessments are conceptual, derived from architectural analysis of control coverage gaps. Quantitative validation through production deployment monitoring is identified as a priority future research direction.

The comparison illustrates that traditional input-security approaches, while necessary, are structurally insufficient for production agentic deployments. The addition of PIPE-informed controls across Layers 2–5 converts a perimeter-only security posture into a defence-in-depth architecture — a transformation that aligns with the layered security principles articulated in both NIST guidance and enterprise security architecture standards. Critically, the improvement in regulatory alignment from partial to full coverage of NIST AI RMF functions has direct compliance value for organisations subject to audit under emerging AI governance regimes.

6. Conclusion

Prompt injection in LLM agent pipelines is not a single, well-bounded vulnerability — it is a family of related threats distributed across distinct architectural layers, each with its own propagation logic, persistence characteristics, and enterprise impact profile. The PIPE taxonomy developed in this study provides the first comprehensive, production-grounded enumeration of these surfaces, offering security architects a structured instrument for threat modelling that moves well beyond the user-input-centric framing that dominates both practitioner discourse and current governance documentation.

Theoretically, the study advances understanding of agentic AI security by establishing that the memory subsystem and inter-agent communication layers represent qualitatively novel threat surfaces — not merely extensions of existing prompt injection requiring purpose-built detection and response architectures. The persistence and cross-session propagation characteristics of memory-layer attacks, in particular, demand a reconceptualisation of the attacker model appropriate to agentic deployments.

From a policy standpoint, the finding that Layers 3 and 5 currently lack standardised control specifications is a direct call to action for standards bodies. The NIST AI RMF and OWASP LLM Top 10 provide essential foundations, but neither addresses inter-agent communication auditing or memory integrity monitoring with sufficient specificity for implementation guidance. Regulators implementing the EU AI Act's technical documentation requirements should explicitly incorporate PIPE-layer coverage as a conformity criterion for high-risk agentic AI systems.

7. Future Scope

Several research extensions follow naturally from this work. Empirical validation of the PIPE taxonomy through controlled red-team exercises against production LLM agent deployments would provide quantitative data on injection success rates and control effectiveness across layers. Longitudinal monitoring studies tracking memory poisoning persistence across extended agent operation periods would establish the practical severity of cross-session attacks. At the governance level, development of layer-specific control specifications particularly for Layers 3 and 5 represents an urgent standards gap that bodies including NIST, ENISA, and ISO/IEC JTC 1/SC 42 are positioned to address. Finally, as multi-agent architectures scale toward larger agent networks with more complex communication topologies, formal verification methods for inter-agent message integrity may become necessary a frontier problem at the intersection of formal methods, cryptography, and AI security research.

REFERENCES

1. Andriushchenko, M., Souly, A., Dziemian, M., Duenas, D., Lin, M., Wang, J., Hendrycks, D., Zou, A., Kolter, Z., Fredrikson, M., Winsor, E., Wynne, J., Gal, Y., & Davies, X. (2025). AgentHarm: A benchmark for measuring harmfulness of LLM agents. arXiv preprint. https://arxiv.org/abs/2410.09024
2. Bandi, A., Kongari, B., Naguru, R., Pasnoor, S., & Vilipala, S. V. (2025). The rise of agentic AI: A review of definitions, frameworks, architectures, applications, evaluation metrics, and challenges. Future Internet, 17(9), 404. https://doi.org/10.3390/fi17090404
3. Beurer-Kellner, L., Buesser, B., Cretu, A., Debenedetti, E., Dobos, D., Fabian, D., Fischer, M., Froelicher, D., Grosse, K., Naeff, D., Ozoani, E., Paverd, A., Tramer, F., & Volhejn, V. (2025). Design patterns for securing LLM agents against prompt injections. arXiv preprint. https://arxiv.org/abs/2506.08837
4. Datta, S., Roy, S., & Chakraborty, T. (2025). Agentic AI security: Threats, defenses, evaluation, and open challenges. arXiv preprint. https://arxiv.org/abs/2510.23883
5. Debenedetti, E., Zhang, J., Balunovic, M., Beurer-Kellner, L., Fischer, M., & Tramer, F. (2024). AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents. arXiv preprint. https://arxiv.org/abs/2406.13352
6. Dong, S., Xu, S., He, P., Li, Y., Tang, J., Liu, T., Liu, H., & Xiang, Z. (2025). A practical memory injection attack against LLM agents. arXiv preprint. https://arxiv.org/abs/2503.03704
7. European Parliament. (2024). Regulation (EU) 2024/1689 — Artificial Intelligence Act. Official Journal of the European Union. https://eur-lex.europa.eu/eli/reg/2024/1689/oj
8. European Union Agency for Cybersecurity (ENISA). (2023). ENISA threat landscape for AI. https://www.enisa.europa.eu/publications/enisa-threat-landscape-for-ai
9. Fu, X., Li, S., Wang, Z., Liu, Y., Gupta, R. K., Berg-Kirkpatrick, T., & Fernandes, E. (2024). Imprompter: Tricking LLM agents into improper tool use. arXiv preprint. https://arxiv.org/abs/2410.14923
10. Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. (2023). Not what you've signed up for: Compromising real-world LLM-

integrated applications with indirect prompt injection. Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec '23), 79–90. https://doi.org/10.1145/3605764.3623985
11. Narajala, V. S., Huang, K., & Habler, I. (2025). Enterprise-grade security for the model context protocol (MCP): Frameworks and mitigation strategies. arXiv preprint. https://arxiv.org/abs/2504.08623
12. National Institute of Standards and Technology. (2023). Artificial intelligence risk management framework (AI RMF 1.0) (NIST AI 100-1). U.S. Department of Commerce. https://doi.org/10.6028/NIST.AI.100-1
13. OWASP Foundation. (2025). OWASP Top 10 for large language model applications (Version 2.0). https://owasp.org/www-project-top-10-for-large-language-model-applications/
14. Raza, S., & Emmanouilidis, C. (2025). TRiSM for agentic AI: A review of trust, risk, and security management in LLM-based agentic multi-agent systems. arXiv preprint. https://arxiv.org/abs/2506.04133
15. Zou, A., Wang, Z., Kolter, J. Z., & Fredrikson, M. (2023). Universal and transferable adversarial attacks on aligned language models. arXiv preprint. https://arxiv.org/abs/2307.15043