Carnegie Mellon University
Software Engineering Institute

SOK: BRIDGING RESEARCH AND PRACTICE IN LLM AGENT SECURITY

Keltin Grimes, Julie Lawler, Robert C. Garrett, Emil Mathew, Marco Christiani, Sara Kingsley, Zhiwei Steven Wu, Nathan VanHoudnos

November 2025

DOI: 10.1184/R1/30610928

Abstract

Large Language Model agents are rapidly transitioning from research prototypes to deployed systems, raising new and urgent security challenges. Unlike static chatbots, LLM agents interact with external tools, data, and services, creating pathways to real-world harm even during early stages of development. Existing guidance on securing agents is fragmented, creating obstacles for developers and organizations looking to build secure systems. To clarify the security landscape, we conduct a systematic review covering academic surveys, grey literature sources, and real-world case studies. We then (i) categorize the known threats to LLM agents and analyze key attack surfaces, (ii) construct a taxonomy of actionable security best practices encompassing the full LLM agent development lifecycle, highlighting gaps in the security landscape, and (iii) evaluate the adoption of these recommendations in practice. Together, these contributions establish a framework for developing comprehensive risk-mitigation strategies. Our synthesis promotes standardization, surfaces gaps in current practice, and establishes a foundation for future work toward secure LLM agents.

Introduction

The growth of Large Language Models (LLMs) from text-only chat interfaces to agentic systems has created pressing security concerns. Unlike their predecessors, LLM *agents* are designed to perceive their environment and take actions in the real world, providing a plethora of new attack surfaces and avenues to harm. The risks continue to escalate as LLM agents become increasingly autonomous [1] and integrated into sensitive applications such as healthcare [2, 3, 4], cybersecurity [5, 6, 7], finance [8, 9, 10], and energy [11, 12, 13].

Importantly, the security risks of LLM agents have 'shifted left' [14], emerging much earlier in the development lifecycle than for conventional LLMs. Standalone LLMs typically face concerns such as content safety,

privacy leakage, or harmful outputs only after deployment, once external users interact with them [15]. In contrast, LLM agents begin to face security threats as soon as they are connected to external tools, data, or other agents–even in prototype stages. For instance, enabling web search in an early experiment can already expose the agent to prompt injection attacks [16, 17], while connecting to an untrusted remote tool server could allow arbitrary code execution on the developer's machine [18]. These examples illustrate that security is not just a production concern; even researchers with no intention of deployment must consider securing their systems against these new threats.

The expanded risks of LLM agents have catalyzed growing concern from researchers across academia [19, 20], industry [21, 22], and government [23, 24]. Despite this widespread concern, **standard practices for securing LLM agents remain fragmented.** Much of the progress in both research and deployment experience comes from industry, but contributions are often tied to proprietary ecosystems or commercial incentives, limiting their generalizability and objectivity. Academic work has produced a growing set of surveys and threat taxonomies, but tends to emphasize future directions over practical guidance. A consensus-driven synthesis of security best practices that spans both research and practice is essential to translate emerging insights into actionable guidance.

To address this gap, we systematize knowledge across academia, industry, and real-world deployments to provide a unified framework for the secure design, development, and operation of LLM agents.

1. We **categorize threats to LLM agents** mentioned across our corpus, identifying which attack surfaces receive the most attention. This contribution establishes the most pressing threats and highlights differences in perspectives between academia and industry.
2. We introduce a comprehensive **taxonomy of security best practices** covering the development and deployment lifecycle of LLM agents. This contribution presents both a practical reference for developing secure LLM agents and a framework for identifying understudied areas that require further development.
3. We analyze security techniques used in real-world LLM agent case studies to **evaluate adoption in practice**. This contribution highlights discrepancies between recommendations in the literature and deployed systems.

Building on these contributions, we conclude with three recommendations for advancing security research on LLM agents, including a demonstration of how our framework can be applied to conduct systematic risk assessments.

Preliminaries and Related Work

This work focuses on the security of *LLM agents*, AI systems whose primary perception, reasoning, and decision-making module is an LLM. While definitions of an agent vary across contexts [25, 26, 27], we follow the definition provided in the work of Chan and colleagues [28], who characterize agency in algorithmic systems through four key properties: accomplishing underspecified objectives, acting directly in the world without human mediation, exhibiting goal-directed behavior, and conducting long-term planning. We adopt this framing because it is not tied to any particular technique or feature, and it emphasizes precisely the

capabilities that raise new security concerns. Our scope is LLM-based systems with meaningfully more agency than a static LLM chatbot [29]. For simplicity, we frequently refer to LLM agents as "agents."

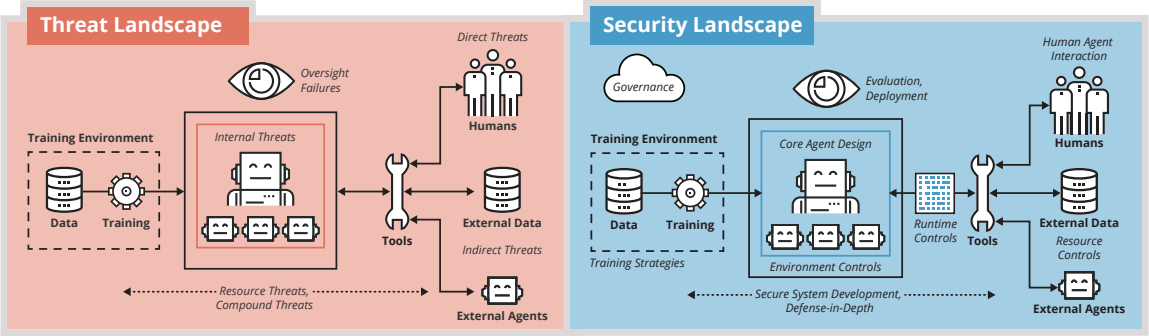

Figure 1: A generalized reference architecture of LLM agents. Agent components are displayed as icons and/or labeled in bold. We map our taxonomy of threat surfaces (left) and taxonomy of security best practices (right) to the reference architecture. Taxonomy categories are italicized.

In practice, a core LLM reasoning and planning engine exercises agency through software scaffolding that connects it to external tools. These tools enable perception of the environment, execution of actions, and delegation of tasks to specialized systems or other agents. Techniques such as reinforcement learning on long-horizon, multi-step tasks [30] strengthen an agent's capabilities, allowing it to operate coherently for periods beyond those of a static LLM [1].

In Figure 1 we present a schematic of an LLM agent, representative of prevailing LLM agent architectures, showing the core (multi-)agent engine, training environment, tools to interact with external entities, and independent oversight mechanisms. On the left, we overlay our taxonomy of threats to LLM agents introduced in Threats to LLM Agents, and on the right, we overlay our security best practices taxonomy presented in Security Best Practices, both italicized. This dual mapping shows how threats and mitigations correspond to specific components of the agent architecture.

Throughout the paper, we follow standard cybersecurity definitions for key terminology to promote clarity and standardization between AI security and cybersecurity [31]. We frequently refer to definitions from the Committee on National Security Systems Glossary [32]. We define a "threat" to be "any circumstance or event with the *potential* to adversely impact organizational operations, organizational assets, individuals, [or] other organizations," while "attacks" are actions taken along those lines [32]. "Risk" is "a measure of the extent to which an entity is threatened by a potential circumstance or event" and is measured by the magnitude of harm and the likelihood of occurrence [32].

Comparison to Related Work

As part of our systematic literature review, we conducted a meta-review of academic surveys on LLM agent security (described in Methods). Much of the prior work limited its scope to a particular application area (e.g., [33, 34, 35]), risk (e.g., [36, 37, 38]), or mitigation technique (e.g., [39, 40, 41]), and by design covered only a subset of the security landscape. Fewer than a third of the papers we analyzed detailed a repeatable literature review methodology. Only three papers explicitly reviewed grey literature and only five analyzed real-world LLM agent case studies. We are the first to conduct a meta-review of prior review papers. We provide a detailed comparison with all surveyed academic work in the appendix.

Notably, no single source from any of the reviewed literature was sufficient to cover either the threat landscape

or the security landscape according to our taxonomies. Huang's work on MAESTRO [42] described attacks in 19 of the 25 classes of attacks noted in Threats to LLM Agents, but no other sources covered more than half and only seven covered more than a third. Of the 33 categories of security controls described in Security Best Practices, only two sources [43, 44] provided recommendations in more than half of the categories (each with 19); only five reached more than a third. This further emphasizes the utility of our work in providing a complete picture of the security ecosystem. We provide a more detailed breakdown of these distributions in the appendix.

Some documents also referenced existing threat modeling frameworks, which collect and organize threat intelligence to inform threat modeling, especially for red-teaming [45, 46]. The framework most similar to our work is MAESTRO [42], a recent agent-specific framework whose categorized threats span all seven attack surfaces in our threat taxonomy. However, MAESTRO does not cover six of the 25 threat categories we identified, excluding notable threats like indirect prompt injection and adverse multi-agent dynamics. These frameworks also do not present the prevalence of particular threats in the literature, which our review does.

Methods

We now detail our systematic literature review process, which followed the methodology described in the work of Siddaway and colleagues [47].

We collected academic literature and grey literature sources separately. The grey literature was divided into general industry guidance and case studies of real-world agents. Sources from academia and industry serve as our baseline for identifying best practices and threats, while case studies serve as our point of comparison between recommendations and actual deployments. For each source category, we provide procedural details, calibration testing results, and counts of documents included at each stage. Detailed search strings for all sources are provided in the Search Strategy Details. We also describe our methods for categorizing and analyzing the collected data.

Academic Literature

The academic literature review focused on reviews, surveys, systematizations of knowledge (SoKs), and similar overview papers to determine current areas of research on LLM agent security. The goal was to identify studies covering agentic or tool-using LLM systems and to extract threats and actionable security guidance.

Initial Search Strategy

We queried the Semantic Scholar search API [48] using combinations of search terms targeting study scope, LLM agent focus, and security content within a five-year window (July 2020–July 2025), restricted to computer science. A total of 100 searches were conducted and results were sorted by built-in relevance ranking. The top 15 results per query were collected, and de-duplication produced a pool of 478 candidate papers.

Title and Abstract Screening

All 478 papers were screened at the title level using inclusion criteria focused on (i) mention of AI, LLMs, or agents and (ii) safety or security relevance. At this stage, two rounds of calibration testing were conducted to assess reviewer agreement. In the final round, five reviewers each screened 10 papers; 9/10 had full agreement and 1/10 had 4/5 agreement. Title screening yielded 209 papers for the subsequent abstract screening step.

Abstract screening applied the following criteria: (i) the paper is a review, survey, SoK, taxonomy, or other multi-technique analysis; (ii) the focus is agentic or tool-using LLM systems or similar advanced capabilities; (iii) the paper includes recommendations or open problems related to secure agent design. At this stage, two additional rounds of calibration testing were conducted. In the final round, 3 reviewers each screened 6 papers; 5/6 had full agreement and 1/6 had 2/3 agreement. Abstract screening yielded 64 papers for extraction.

Industry Literature and Case Studies

To complement the academic review, we reviewed the grey literature to identify industry guidance (e.g., industry reports, white papers, corporate blog posts) and case studies (e.g., deployments, product or system write-ups). Guidance for our grey literature searches follows [49].

Initial Search Strategy

Industry literature was targeted via broad automated Google search queries, spanning content type, agent focus, and security terms, with common academic and social media domains excluded. For each query, up to the top 20 results were recorded, yielding 765 candidate sources from industry.

Case studies were seeded from the AI Agent Index [29] (coverage to January 2025) and extended via targeted searches. Relevant documents were identified by manually reviewing links from the AI Agent Index (both the linked documentation and developer websites) and by targeted Google search queries restricted to developer websites. Up to 50 results per website were reviewed. This yielded 154 candidate case study sources.

Source Screening and Separation

Both sets of searches yielded a mix of general industry guidance and specific case studies; the results were partitioned during the screening process to eliminate overlap. Sources describing a specific system, deployment, or project were designated as case studies. Sources not directly tied to a specific system were classified as industry literature.

First, the set of sources seeded from the AI Agent Index [29] were screened in full, due to the index's narrower scope and the use of manual searching methods. We required sources to (i) focus on agentic or tool-using LLM systems, and (ii) have security-related content. Sources not associated with case studies were moved to the industry literature pool. Sixty three case study documents, including two added from the industry pool, advanced to the extraction phase, representing 41 unique case studies in total.

Industry screening mirrored the academic process, but added additional criteria to account for the more variable source quality [50]. Title screening followed the same criteria as the academic sources, but also excluded social media, forums, documentation-only pages, and academic papers. Any case studies in the industry pool were separated at this stage. Two rounds of calibration testing were completed. In the final round, five reviewers each screened 10 documents: 9/10 had full agreement and 1/10 had 4/5 agreement. A total of 349 documents advanced to the abstract stage.

For abstract screening, we reviewed any available abstracts, executive summaries, tables of contents, and section headers, following guidance from the work of Godin and colleagues [50] to account for the lack of standardized abstracts. Criteria included (i) verifiable date and author with technical role or reputable publisher; (ii) focus on tool-using LLMs, agentic systems, or advanced AI; (iii) presence of vulnerabilities, mitigations, or best practices in a security context; and (iv) substantive discussion of recommendations beyond brief mentions. Two rounds of calibration testing were performed. In the final round, three reviewers screened nine documents. Full agreement was reached for 6/9 documents and 3/9 documents had 2/3 reviewer agreement. A total of 109 industry sources advanced to the extraction phase.

Data Extraction

To extract threats and best practices, we used a two-round full-text extraction process. Starting with the 64 academic studies and 109 industry sources, we extracted definitions, threats, and threat models for LLM agents and multi-agent systems. We flagged studies for best practice extraction based on: (i) presence of explicit practices or mitigations for secure LLM agents and (ii) substantive discussion of practices beyond brief mentions. This yielded 24 academic studies and 31 industry documents for the second stage. For each best practice, we recorded: (i) the original description, (ii) a broad practice category, (iii) addressed threats, (iv) use cases, and (v) limitations. We did not merge identical recommendations across sources, allowing us to capture how frequently each practice was recommended. A total of 248 and 258 best practices were recorded from academic and industry sources, respectively.

From the 41 case studies, we extracted security practices implemented for the system. We recorded (i) the original description and (ii) a broad practice category. 13 documents, including five complete case studies, were removed due to insufficient security content, leaving 36 total case studies. A total of 235 security practices were documented.

Categorization of Threats and Best Practices

Following extraction, we aggregated all of the threats collected from the academic and industry sources (64 and 109 documents, respectively). We then normalized identical concepts and assigned initial categories to each threat. We iteratively refined the categories, grouping threats into corresponding threat surfaces (see Fig. 1), until all entries were covered by an appropriate category. Within each threat surface, we organized the threats into specific classes of vulnerabilities.

A similar process was followed for the security best practices extracted from the final set of sources (24 academic, 31 grey literature). The top-level categories correspond to stages of the AI system development lifecycle [51]. Mid-level categories map to particular components of the agent architecture (Fig. 1) or standard processes such as evaluation or deployment. Leaf categories group specific categories of techniques or recommendations.

Categorization of Case Studies

After the best practice taxonomy was finalized, we categorized the security controls reported in case studies against it. Specifically, for each extracted security measure in the case study corpus, we cross-referenced it with the best practice category descriptions and recommendations in each category to determine a best fit. All implemented controls fit an existing category, allowing us to directly assess how the taxonomy categories

are utilized in practice. The results are discussed in Case Studies.

Assessing Selected Threats

We selected two examples of threats from our threat taxonomy to conduct a risk assessment for (Discussion, **R**3). For each threat, we identified vulnerable components in our reference agent architecture (Fig. 1). We then traced the components back to categories in our best practice taxonomy to identify potential security controls. We analyzed the recommendations in each category, as well as the corresponding entries in the case studies, to assess what security controls were applicable. We then condensed the results into a single threat-specific mitigation plan.

Threats to LLM Agents

We present our taxonomy of the threat landscape of LLM agents and provide a comparative analysis to highlight which threats are emphasized across academia and industry.

Threat Taxonomy

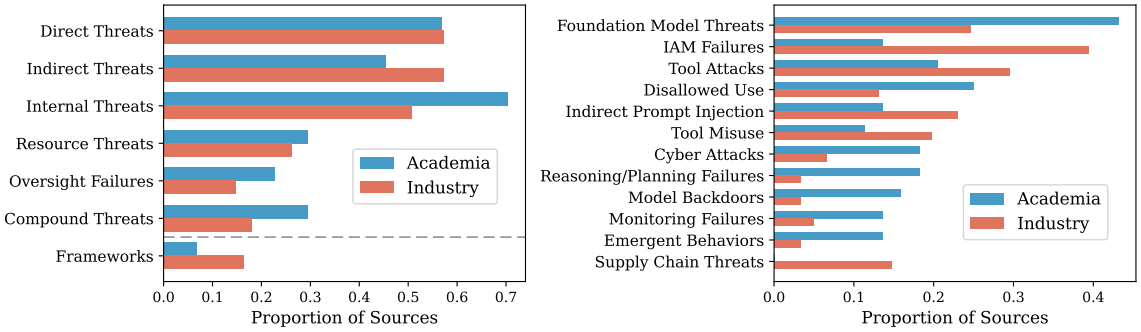

Figure 2: Left: Proportion of sources that mentioned a threat from each threat surface. Proportions were calculated separately for academia and industry. References to threat modeling frameworks are included below the dashed line. Right: Proportion of sources mentioning 12 specific threats. These threats were chosen because they exhibited the largest difference in proportion between academia and industry. Sources that did not mention any threats were excluded.

Each threat is categorized first by threat surface, then by vulnerability or attack type. We include threat modeling frameworks as an additional category to capture their usage. The threat surfaces are listed in Table 1, and overlaid on our reference architecture of LLM agents in Figure 1. A selection of attack types is shown in Figure 2 (right). Here, we describe each threat surface and its associated attack types. Table 1 also includes the lists of sources used to inform the description of each threat surface. Each category is annotated with (n=X, 🎓=Y, 🏭=Z), which represents the number of sources mentioning threats in that category from the complete set of sources (X), academia (Y), and industry (Z).

Table 1: Sources referenced for each threat surface.

Surface	References
Direct Threats	[52, 53, 54, 55, 20, 33, 34, 56, 57, 58, 38, 35, 59, 60, 61, 62, 63, 64, 65, 66, 67, 19, 68, 69, 70, 71, 72, 44, 21, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 43, 102]
Indirect Threats	[52, 54, 55, 20, 34, 58, 38, 35, 59, 60, 103, 104, 65, 105, 106, 107, 68, 69, 70, 108, 72, 44, 73, 109, 74, 75, 76, 22, 110, 111, 112, 113, 114, 79, 115, 116, 80, 117, 84, 118, 119, 120, 121, 89, 91, 92, 122, 94, 123, 97, 23, 98, 99, 124, 102]
Internal Threats	[52, 53, 125, 54, 20, 126, 127, 40, 34, 57, 128, 129, 130, 38, 35, 59, 60, 131, 61, 62, 104, 64, 65, 132, 105, 66, 67, 133, 107, 68, 70, 71, 21, 73, 74, 75, 110, 112, 114, 79, 115, 80, 82, 83, 134, 86, 87, 135, 91, 122, 136, 94, 95, 96, 123, 97, 137, 98, 100, 124, 43, 138]
Resource Threats	[139, 125, 55, 20, 33, 34, 130, 38, 35, 59, 66, 107, 68, 72, 73, 22, 110, 111, 112, 79, 116, 135, 89, 91, 94, 95, 98, 99, 102]
Oversight Failures	[52, 40, 33, 34, 39, 129, 35, 131, 104, 19, 75, 110, 112, 84, 134, 86, 91, 122, 94]
Compound Threats	[20, 33, 38, 59, 103, 131, 62, 67, 107, 19, 68, 70, 108, 44, 75, 110, 112, 115, 90, 91, 94, 95, 124, 43]
Frameworks	[140, 104, 105, 141, 112, 80, 83, 119, 90, 91, 142, 143, 102]

Direct Threats (n=60, 🎓=25, 🏭=35)

Direct threats occur at the application level where the user interacts with an agent. User inputs to the system create vulnerabilities from direct prompt injection (n=52), disallowed use (n=19), or direct multimodal attacks (n=2).

Indirect Threats (n=55, 🎓=20, 🏭=35)

Components connected to a core LLM engine introduce additional threats. Content automatically inserted into prompts as context is vulnerable to indirect prompt injection (n=20). Although any component can introduce a new vulnerability, three components were frequently noted as susceptible to attack: tools (n=27), knowledge-bases (n=15), and internal agents (n=15). Specific threats include memory poisoning, insecure plugin design, and communication protocol exploits. Identity and access management (IAM) controls can be implemented to secure the agent system (Access Controls), but remain vulnerable to misuse or attack (n=30).

Internal Threats (n=62, 🎓=31, 🏭=31)

The LLM engine that powers an agent's perception, decision-making, and actions may itself introduce internal threats. These vulnerabilities can arise during development, such as through data poisoning (n=14) or model backdoors (n=9), as well as during deployment. Foundation model vulnerabilities (n=34), including misalignment, hallucinations, and inaccuracy, are a consistent threat. Additional risks stem from an agent's analytical processes, such as planning and reasoning failures (n=10) and deception or evasion behaviors (n=10). Finally, the introduction of tools carries the risk of tool misuse (n=17), where an agent deliberately or inadvertently uses tools in a disallowed manner.

Resource Threats (n=29, 🎓=13, 🏭=16)

These attacks target the underlying software and hardware infrastructure that supports an agent. They include compromise of physical devices (n=4) or cyber components (n=12), misuse of computational resources (n=13), and supply chain attacks on software dependencies integrated into the agent (n=9).

Oversight Failures (n=19, 🎓=10, 🏭=9)

In some agent systems, users can review the agent's reasoning and approve actions. However, this oversight can introduce new vulnerabilities by fostering over-trust through unfaithful explanations (*explainability fail-*

ures, n=9), overloading reviewers (*human-in-the-loop failures*, n=8), or missing issues due to coverage gaps or drift (*monitoring failures*, n=9).

Compound Threats (n=24, 🎓=13, 🏭=11)

The iterative and interactive nature of LLM agents creates new threat surfaces. The iterative loops agents use to execute multistep tasks create cascading failures (n=14) where failure in one step is magnified in successive steps. Systems interacting with other autonomous agents are susceptible to adverse multi-agent dynamics (n=13) (e.g., coordination failures or multi-agent drift, among others). The broader architecture of an agent, including its design and constituent components, can lead to system-level failures (n=5).

Frameworks (n=13, 🎓=3, 🏭=10)

Some documents also referenced existing threat modeling frameworks. Occasionally, these were referenced instead of enumerating specific threats. The top frameworks were: MAESTRO (n=6), a recent agent-specific framework [42]; OWASP Top 10 for LLMs (n=5) [144]; MITRE ATLAS (n=3) [145]; and STRIDE (n=2), a classic cybersecurity framework [146].

Discussion of the Threats

Takeaway 1: *Traditional cyber threats are a growing concern.* The traditional risks of LLMs center on threats specific to the model itself, with little concern for cyber-focused threats [15], reflecting their limited integration with external systems. However, the literature on LLM agents points to traditional cybersecurity threats as a major concern for these AI systems. Well-known cyber threats such as denial-of-service, man-in-the-middle, and code injection attacks were repeatedly points of concern. This provides evidence to support the growing trend of drawing on existing software security principles to inform the security of AI systems [147, 51, 148, 149, 31].

Takeaway 2: *Academia focuses on model-centric and emerging threats.* Academic work places additional emphasis on internal threats, especially threats relating directly to the foundation model, such as misalignment and hallucination, as well as reasoning and planning failures, backdoor attacks, and emerging threats such as deception and scheming. In contrast, industry sources concentrate on threats resulting from system integration, especially indirect threats, IAM failures, attacks on and misuse of tools, indirect prompt injection, and supply chain threats. These emphases reflect differences in incentives—academic teams focus on novel and emerging failure modes, while industry teams prioritize operational threats in deployed systems. The two lenses are complementary, supporting a more complete taxonomy of agent threats, but also suggest opportunities for the academic community to engage more with system-level concerns.

Takeaway 3: *Threat modeling needs more standardization.* Structured threat taxonomies help practitioners compare systems, prioritize mitigations, and accumulate evidence across studies by providing shared structure and terminology [150, 45, 151]. In our review, references to existing taxonomies were sparse and fragmented across frameworks. The terminology used to describe threats varied widely, and we observed a long tail of terms describing the same or very similar threats. Coalescing around a shared set of frameworks and terminology would help translate insights between research and practice, and between cyber and AI security.

Security Best Practices

Our taxonomy of security best practices (Figure 3) is split into Design, Development, and Operations, covering the three stages of the LLM agent lifecycle. We define each leaf category and describe the core recommendations. In Table 2, we cite the lists of sources used to inform the description of each category. As with the threat taxonomy, each section is annotated with (n=X, 🎓=Y, 🏭=Z), representing the number of sources recommending best practices in that category from the complete set of sources (X), academia (Y), and industry (Z).

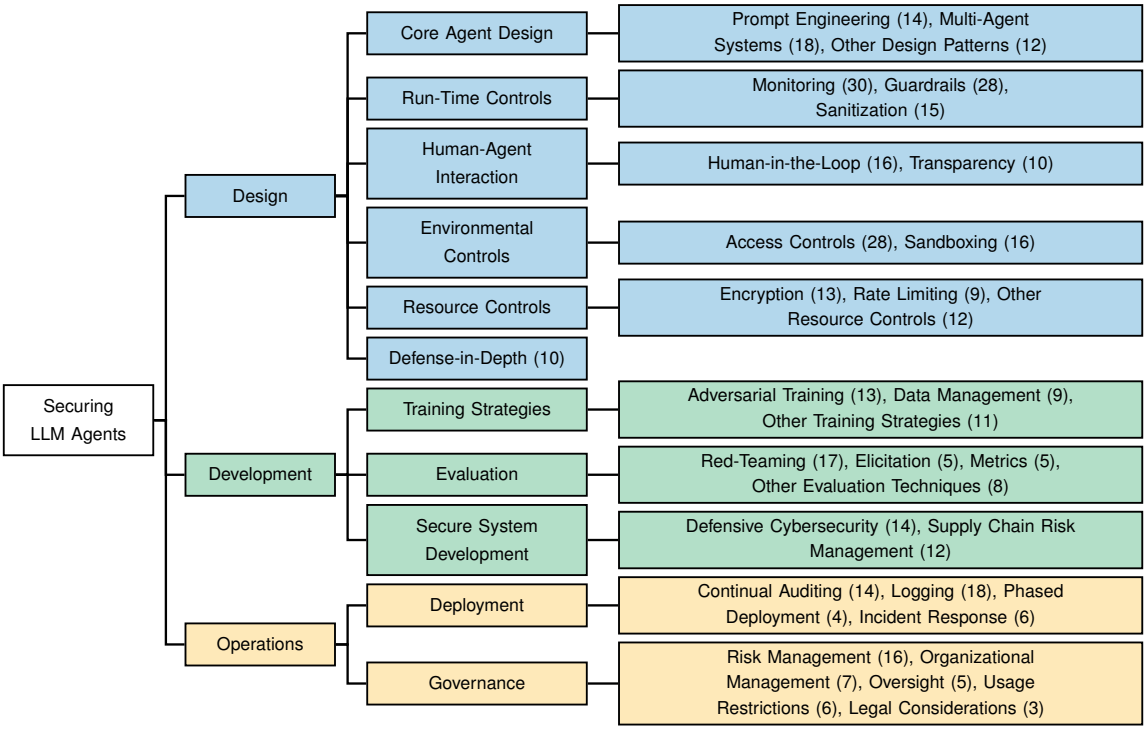

Figure 3: Taxonomy of best practices for LLM agent security. Each leaf is annotated with the number of documents providing a best practice in that category. A total of 55 documents from the academic and industry literature reviews are represented here. Color fill is determined by stages of the LLM agent lifecycle.

Design (n=50, 🎓=21, 🏭=29)

Design covers the architecture of the agent and its supporting systems and security controls.

Core Agent Design (n=26, 🎓=16, 🏭=10)

Core agent design refers to the system-level architecture, focusing on how the LLM is integrated and how it interacts with other components, rather than model-level details.

Prompt Engineering (n=14, 🎓=9, 🏭=5)

Prompts customize LLMs for specific applications. The literature recommends various techniques to engineer prompts to promote resilience against threats. This defensive prompting approach emphasizes providing explicit, unambiguous instructions, and clearly demarcating data retrieved by tools to indicate untrusted content.

Multi-Agent Systems (n=18, 🎓=14, 🏭=4)

Multi-agent designs leverage redundancy, critique, and consensus to reduce error and constrain unsafe behavior. Independent agents propose solutions and cross-validate outputs to reduce hallucinations, while multi-agent debates help surface errors and block harmful actions. However, multi-agent setups are resource-intensive and in some cases can amplify security concerns.

Other Design Patterns (n=12, 🎓=5, 🏭=7)

We noted three other design categories that are cited less often: control flow techniques (n=8), grounding (n=5), and continual learning (n=2). Control flow includes techniques that influence the sequence (or flow) of actions or steps that an agent makes. Recommendations included implementing planning-centric designs and using deterministically controlled action sequences when appropriate. Grounding involves leveraging trusted sources of information, such as RAG databases, web search, or knowledge graphs, often as a defense against hallucination. Continual learning can offer security benefits by refining model robustness over time.

Run-time Controls (n=44, 🎓=20, 🏭=24)

Run-time controls monitor an agent's inputs and outputs and intervene when issues are detected.

Monitoring (n=30, 🎓=13, 🏭=17)

Many different security concerns can be monitored by examining the inputs, intermediate steps, tool calls, and outputs of an agent. Key techniques include LLM-based classifiers, anomaly detection, communication network-level monitoring. Various concerns were noted about false positives, latency, cost, and adaptive adversaries.

Guardrails (n=28, 🎓=14, 🏭=14)

If monitors flag unwanted or anomalous behaviors, guardrails can block or refuse the corresponding inputs or outputs. As with monitors, guardrails can help address many different security concerns, though face similar limitations. Most sources described blocking or refusing the entire interaction, but some suggested filtering sub-components such as particular tool calls.

Sanitization (n=15, 🎓=11, 🏭=4)

Sanitization techniques seek to neutralize malicious, biased, or sensitive content before it reaches or leaves an LLM agent, rather than blocking or refusing outright. Methods such as paraphrasing, retokenization, masking, and random perturbations were commonly suggested. However, the utility of the agent may degrade with aggressive sanitization.

Human-Agent Interaction (n=20, 🎓=4, 🏭=16)

Human-agent interaction covers techniques to utilize or improve interactions between humans and the agent.

Human-in-the-Loop (n=16, 🎓=2, 🏭=14)

Human-in-the-Loop (HITL) approaches ensure that humans have oversight and control over agent actions. Requiring explicit human confirmation before high-risk or irreversible actions and allowing for the agent's execution to be interrupted at any point were the primary recommendations. Careful design of HITL steps can help balance effective supervision with operator fatigue.

Transparency (n=10, 🎓=3, 🏭=7)

Transparency practices focus on making AI agent decision-making processes interpretable and accountable to users. Sources suggest providing agent-generated explanations for decisions or exposing its intermediate processes directly, and ensuring that user interfaces provide clarity on how user decision may impact agent behavior. Explainability tools should be paired with cognitive forcing mechanisms to ensure user engagement.

Environmental Controls (n=32, 🎓=13, 🏭=19)

Environmental controls mediate the agent's interaction with the external environment.

Access Controls (n=28, 🎓=10, 🏭=18)

Access controls regulate the systems, data, and resources with which agents can interact. All interactions should be authenticated and authorized, using role-based and task-specific permissions. Permissions should be granular and data sources and tool capabilities should be partitioned to support this. The principle of least privilege [152] was repeatedly referenced as a guideline.

Sandboxing (n=16, 🎓=7, 🏭=9)

Sandboxing constrains an agent to operate within secure boundaries to minimize exposure to external systems. Isolated execution environments can contain malicious actions by the agent or block external attacks. Sandboxed environments should provide minimal privileges for tool, data, and network access, while still accurately emulating real-world environments.

Resource Controls (n=24, 🎓=13, 🏭=11)

Security measures designed for data or other resources associated with or used by an agent fall under resource controls.

Encryption (n=13, 🎓=6, 🏭=7)

The data used by an agent, as well as any communications it sends over a network, should be encrypted in transit and at rest. Key parts of the agent system itself, including prompts and model weights, should be encrypted to prevent theft. Some sources recommended using homomorphic encryption or secure multi-party computation protocols, but their practical applicability is unclear.

Rate Limiting (n=9, 🎓=4, 🏭=5)

Rate limiting restricts the amount of communication, data, or other resources available to an agent. Limits should be set on run time, CPU usage, data storage, and tool usage to prevent overuse. Rate limits can be set dynamically based on behavioral or historical patterns to better allocate resources.

Other Resource Controls (n=12, 🎓=8, 🏭=4)

Three additional categories of resource controls were observed: watermarking (n=5), privacy controls (n=4), and memory controls (n=3). Watermarking embeds a signature into a piece of data to track its provenance [153, 154, 155], and can be applied to both model outputs and model weights.

The privacy controls mentioned mainly included standard measures such as minimizing record retention and local data processing. For memory controls, two sources recommended performing periodic scans of agent memory systems to block or modify malicious memory entries, and one source recommended limiting how long information persists in memory and giving users control over what information is stored.

Defense-in-Depth (n=10, 🎓=2, 🏭=8)

Defense-in-depth is a security strategy that "involves layering heterogeneous security technologies ... to ensure that attacks missed by one technology are caught by another" [156]. This principle was cited by name in six sources, and others used related terminology such as 'layered security controls.' Though defense-in-depth can encompass security measures from the entire taxonomy, we place it under Design to emphasize that this approach should be taken seriously from the beginning.

Development (n=45, 🎓=19, 🏭=25)

Development covers the training of the agent or the underlying LLM, the software development of the agent scaffolding, and the evaluation of the system.

Training Strategies (n=20, 🎓=13, 🏭=7)

This category covers security-focused training strategies and techniques for securing the corresponding training data.

Adversarial Training (n=13, 🎓=9, 🏭=4)

Adversarial training involves exposing a model to adversarial attacks and training it to be robust to them. The literature recommends adversarial training for a variety of security threats beyond traditional adversarial examples, such as prompt injection and multi-agent collusion.

Data Management (n=9, 🎓=7, 🏭=2)

To manage training data-related threats, the literature primarily recommended filtering corrupted or poisoned data from training datasets to prevent backdoor attacks, and applying differential privacy or other privacy-preserving techniques to reduce privacy leakage. Some sources emphasized data diversity to avoid model bias.

Other Training Strategies (n=11, 🎓=7, 🏭=4)

Various other training strategies were suggested; none appeared in more than three sources. These techniques included Reinforcement Learning with Human Feedback (RLHF), a popular approach to aligning LLMs with human preferences; Machine Unlearning, a family of techniques for removing certain data or behaviors from trained ML models; and instruction hierarchy training, which involves training models to prioritize instructions from certain sources over others [157], primarily to defend against prompt injection attacks.

Evaluation (n=23, 🎓=10, 🏭=13)

Evaluations assess the extent to which an LLM agent and its security controls meet the desired security properties.

Red-Teaming (n=17, 🎓=7, 🏭=10)

Red-teaming involves emulating potential attacks against a system to identify security vulnerabilities and highlight opportunities for defensive improvements [32]. For agents, the literature emphasizes red-teaming the entire threat surface and all system components, covering a variety of security concerns, and viewing red-teaming as a continual process where attacks are used to refine both defenses and attack creation. Red teams should leverage previously observed real-world attacks, use automated attack generation and red-teaming tools, and emulate realistic adversaries under realistic conditions.

Elicitation (n=5, 🎓=3, 🏭=2)

Capability elicitation is the process of drawing out the full range of behaviors or skills an agent can exhibit, in order to accurately assess its security risks. When evaluating LLM agents, especially for the potential misuse of dangerous capabilities, it is important that elicitation is done carefully to avoid underestimating security concerns. Key techniques include using effective agent scaffolding and following permissive threat models (e.g. white-box access). Evaluators should be able to argue that if attackers develop new attacks or utilize significantly more resources to conduct an attack, security will not be compromised.

Metrics (n=5, 🎓=1, 🏭=4)

Our corpus provided various guidance for designing effective metrics for evaluations, including quantifying harm on granular scales rather than binary pass/fail metrics, aligning metrics with real-world outcomes, and, when the agent is the threat, evaluating both the capacity and the propensity of the agent to cause harm.

Other Evaluation Techniques (n=8, 🎓=3, 🏭=5)

Our sources noted a variety of other recommendations for conducting effective evaluations, including using end-to-end integration tests in realistic environments; other testing strategies such as unit testing, regression testing, and variant analysis; and understanding how an agent's situational awareness (see [158]) may impact test results.

Secure System Development (n=21, 🎓=6, 🏭=15)

Secure system development covers security practices embedded in the software development and procurement process.

Defensive Cybersecurity (n=14, 🎓=3, 🏭=11)

Defensive cybersecurity involves passive and active cyberspace activities designed to protect data, networks, net-centric capabilities, and other designated systems [32]. Most of the practices encompassed traditional DevSecOps principles [159] including version control, configuration management, vulnerability patching, static and dynamic analysis, and integrated testing. APIs and training and production environments were highlighted as key threat surfaces for agents to be secured. Some sources cited existing secure development standards such as the NIST Secure Software Development Framework [14].

Supply Chain Risk Management (n=12, 🎓=5, 🏭=7)

Supply chain risk management (SCRM) is a "systematic process for managing supply chain risk by identifying susceptibilities, vulnerabilities, and threats throughout the supply chain and developing mitigation strategies to combat those threats" [32]. SCRM should occur throughout the system, including hardware, software dependencies, models, tools, and external data. Specific mitigations include inspecting model parameters for back doors; performing vendor security assessments; and scanning, auditing, and pinning dependencies.

Table 2: Sources referenced for each best practices category.

Design Development Operations

Category	References
Prompt Engineering	[20, 126, 34, 57, 38, 60, 61, 107, 68, 44, 73, 116, 80, 87]
Multi-Agent	[20, 34, 57, 36, 130, 38, 35, 59, 60, 105, 107, 19, 68, 70, 44, 96, 43, 143]
Other Design Patterns	[20, 34, 57, 130, 38, 60, 68, 44, 110, 86, 96, 43]
Monitoring	[20, 34, 36, 130, 38, 35, 59, 61, 104, 107, 19, 68, 41, 72, 44, 21, 109, 160, 76, 116, 80, 84, 86, 90, 95, 96, 137, 101, 43, 143]
Guardrails	[20, 126, 38, 35, 59, 60, 61, 62, 37, 105, 107, 68, 41, 70, 161, 72, 44, 73, 76, 162, 80, 82, 87, 102, 90, 163, 43, 143]
Sanitization	[126, 34, 57, 36, 38, 59, 60, 61, 105, 107, 68, 161, 73, 84, 101]
Human-in-the-Loop	[39, 130, 161, 44, 21, 160, 76, 113, 80, 84, 86, 87, 122, 163, 101, 143]
Transparency	[126, 34, 39, 44, 160, 110, 86, 87, 101, 43]
Access Controls	[126, 34, 36, 130, 38, 62, 104, 105, 107, 68, 161, 72, 44, 21, 109, 76, 110, 162, 113, 80, 82, 84, 86, 87, 98, 101, 43, 143]
Sandboxing	[20, 34, 57, 130, 38, 37, 107, 72, 44, 73, 109, 76, 110, 86, 43, 143]
Rate Limiting	[20, 38, 59, 19, 72, 44, 76, 101, 143]
Encryption	[126, 38, 37, 105, 107, 70, 161, 109, 80, 84, 86, 43, 143]
Other Resource Controls	[126, 57, 59, 61, 62, 37, 68, 70, 44, 80, 86, 143]
Defense-in-Depth	[38, 104, 141, 76, 80, 102, 90, 163, 101, 43]
Adversarial Training	[20, 34, 57, 38, 60, 61, 107, 19, 70, 44, 21, 87, 43]
Data Management	[34, 57, 38, 59, 60, 37, 107, 44, 80]
Other Training Strategies	[20, 57, 35, 60, 61, 37, 68, 44, 160, 137, 43]
Red-Teaming	[40, 130, 38, 35, 59, 61, 37, 44, 141, 162, 80, 86, 87, 102, 90, 98, 43]
Elicitation	[139, 39, 130, 141, 43]
Metrics	[130, 102, 96, 23, 43]
Other Evaluation Techniques	[130, 37, 19, 86, 87, 95, 23, 43]
Defensive Cybersecurity	[20, 130, 105, 72, 21, 109, 76, 164, 86, 102, 98, 101, 43, 143]
Supply Chain Risk Management	[20, 34, 38, 104, 105, 72, 73, 109, 141, 116, 84, 43]
Continual Auditing	[139, 33, 130, 38, 35, 62, 41, 44, 160, 141, 76, 162, 116, 95]
Logging	[20, 34, 62, 104, 19, 41, 161, 44, 109, 76, 110, 80, 82, 86, 87, 122, 101, 143]
Incident Response	[105, 109, 141, 84, 102, 96]
Phased Deployment	[35, 161, 44, 98]
Risk Management	[139, 40, 35, 105, 19, 70, 44, 109, 141, 76, 113, 102, 90, 98, 101, 43]
Organizational Management	[139, 33, 141, 80, 102, 95, 98]
Oversight	[33, 161, 72, 113, 122]
Usage Restrictions	[35, 72, 141, 162, 102, 43]
Legal Considerations	[76, 80, 95]

Operations (n=39, 🎓=25, 🏭=14)

Operations comprises both the deployment of an LLM agent and the governance structures of the organization surrounding the entire development lifecycle.

Deployment (n=33, 🎓=12, 🏭=21)

Following design and development, an agent will be deployed to production. Deployment includes the actual deployment process and the security controls needed to validate its ongoing operation.

Continual Auditing (n=14, 🎓=7, 🏭=7)

This set of recommendations emphasizes that evaluation should be a continuous process throughout deployment, as new concerns can arise due to domain shift, edge cases, or unseen attacks. Continual auditing often relies on logging (below) to collect and analyze agent interactions. Unique agent identifiers can be used to trace failures back to particular systems or users. Audit results should inform governance processes, including risk management strategies, and technical mitigations like guardrails. Governance processes themselves can also be audited.

Logging (n=18, 🎓=6, 🏭=12)

In support of continual auditing, the literature recommends logging as much information about an agent's execution as possible. Logs can also support incident response protocols, oversight tools, or inter-query monitoring. Logs should be immutable and tamper-resistant or tamper-evident. However, extensive logging raises privacy concerns and can overwhelm auditors if not managed effectively.

Phased Deployment (n=4, 🎓=1, 🏭=3)

Four sources argued for a phased approach to agent deployment, where the scope of deployment is expanded iteratively as safety metrics are met in each successive step. This gradually increases risk while providing opportunities to discover new issues and roll back the deployment if needed.

Incident Response (n=6, 🎓=1, 🏭=5)

When security breaches occur, it is important to already have incident response protocols in place to respond rapidly and effectively. Protocols should have clear steps for containment, eradication, and recovery, and the protocols themselves should be tested and maintained.

Governance (n=23, 🎓=7, 🏭=16)

The literature notes a breadth of processes that can be utilized at an organizational level to manage or promote security, beyond the scope of any single agent deployment. Although outside the purview of individual developers, these techniques are recommended for institutions that develop or manage LLM agent deployments.

Risk Management (n=16, 🎓=6, 🏭=10)

Risk management is the "program and supporting processes to manage risk to organizational operations" [32]. Risk management begins with a structured approach to threat modeling, where potential threat actors, attack pathways, vulnerable assets, and loss outcomes are analyzed to estimate risk. This process should be

informed by existing threat taxonomies, emerging research, and evaluation and red-teaming results. Organizations should participate in threat intelligence sharing [46] to inform threat modeling across the community. Threat modeling results should then be used to establish specific risk thresholds that, when exceeded, will trigger proportional mitigation requirements. Risk thresholds and management policies should be well documented and communicated to internal and external parties to receive feedback and promote transparency and accountability. It is recommended to align with existing security frameworks, regulatory requirements, and governance bodies.

Organizational Management (n=7, 🎓=2, 🏭=5)

Organizational management covers the management of the structure and workforce of the organization. To increase resilience against rapid changes in job responsibilities or the security landscape caused by widespread agent integration, organizations should invest in training workers in AI literacy and security recommendations. Security teams in particular should be kept up-to-date on emerging attack vectors and red-teaming best practices. Leadership should promote a strong 'safety culture' that encourages people to prioritize safety and raise concerns about security. Organizations should also define AI governance policies and usage guidelines to align expectations across the organization, informed by deployment experience and various stakeholders.

Oversight (n=5, 🎓=1, 🏭=4)

Oversight controls are organizational-level tools for standardizing and enforcing security policies. Technical safeguards such as monitoring, guardrails, and access controls can be enforced through centralized systems.

Usage Restrictions (n=6, 🎓=1, 🏭=5)

In addition to access controls for agents, similar restrictions are needed for humans. User access controls limit access to agents to prevent malicious use disallowed by usage policies. Internal developer access to critical agent sub-components, such as model weights or user data, should be minimized and protected with techniques like multifactor authentication to defend against theft. Additional specific technical recommendations show significant overlap with agent access controls.

Legal Considerations (n=3, 🎓=0, 🏭=3)

Three industry sources highlighted certain legal considerations that may be important when developing or deploying agents, such as complying with data protection laws, protecting intellectual property with patents and copyrights, and using indemnification clauses to clarify liability.

Discussion of the Best Practices

Takeaway 4: *The security landscape is broad but permeable.* Our taxonomy, including the literature it covers, provides a wide array of practical techniques that can improve the security of agents. Many categories had solid agreement across academia and industry. Although all have limitations, a defense-in-depth approach can provide substantial security benefits. However, the emphasis on defense-in-depth simultaneously underscores the reality that agents, as with all software, are fundamentally insecure. Layering as many safeguards as possible is the best approach to an unforgiving and rapidly evolving threat landscape.

Takeaway 5: *LLM agent security builds on a large body of cybersecurity knowledge.* Throughout the literature, we found repeated references to foundational techniques in software security and extensions of existing cybersecurity guidance to agents. Many of the best practices in the categories of defensive cybersecurity,

supply chain risk management, and encryption could be applied to almost any software system, regardless of the presence of AI components. As noted by some sources [164, 143], these cyber-focused recommendations are well covered (and expanded upon) by existing guidance such as the NIST Secure Software Development Framework [14] and the OWASP Application Security Verification Standard [165]. Other categories adapt common cybersecurity tools and principles to AI-specific components. Access controls, for example, adapt IAM roles and OAuth protocols [166] to provide agents with their own identities and access restrictions [84, 101].

Furthermore, many of the other categories, although focused on AI-specific components, are closely related to traditional cybersecurity techniques. Monitoring, for example, focused almost exclusively on monitoring the LLM agent itself, despite monitoring being a core component of cybersecurity defenses [167, 14]. The observed red-teaming recommendations also focused on securing AI components against AI-specific threats, although it has a long history in being applied to traditional software components and cyber threats [168, 45]. This suggests further opportunities to leverage the extensive experience of the cybersecurity community in traditional software to improve security controls for LLM agents.

Takeaway 6: *Industry emphasizes operational security considerations and traditional cybersecurity practices.* Industry sources dominated the guidance for eight of the nine leaf categories in the operations stage. This highlights structural differences between the strengths of academia and industry. Industry guidance is often directly informed from deployment experience (e.g. [102, 141, 163]), experience which researchers in academia can generally not obtain directly. For example, the two articles providing the bulk of the academic guidance on LLM agent governance drew heavily on interviews with public sector workers [139] and existing industry and government practices [33]. The industry literature also constituted much of the guidance on human-agent interaction, especially for HITL. This emphasis on user-centric design again highlights the focus on deployment in industry, where agents will interact with real-world users.

We previously highlighted industry's concern with cyber threats in Threats to LLM Agents, and the distribution of best practices continues this trend. The majority of the defensive cybersecurity recommendations came from industry sources (Defensive Cyber), and while the distribution for SCRM was more balanced, industry sources tended to focus more on the traditional software supply chain rather than AI-specific components. This gap could be filled through closer collaboration with academic cybersecurity researchers; recent work on hybrid cyber-AI vulnerabilities demonstrates the utility of leveraging cybersecurity experience [169, 170, 171].

Takeaway 7: *Common limitations of security controls pose feasibility challenges.* Across the entire best practices taxonomy we saw recurring tradeoffs in security controls. The literature noted increased costs [20, 160], lower agent performance [60, 61], and added privacy concerns [34, 160] for a variety of controls, tradeoffs that would only compound with defense-in-depth approaches. For example, some sources recommended minimizing the amount of data stored from agent interactions [62, 86] to address privacy concerns, contradicting other guidance that recommended logging as *much* data as possible [34, 62, 161, 44, 80]. Risk management techniques are designed to guide developers through the difficult decisions posed by conflicting requirements [172], but they still pose challenges for practical adoption. Improving the Pareto frontier of these tradeoffs is a promising direction for future work.

Case Studies: Security Controls in Practice

In this section, we present and analyze the results from our review of LLM agent case studies, which provide insight into the real-world usage of agent security controls.

Case Study Results

We list the number of case studies that described a security control in each category of our best practice taxonomy. Counts are out of 36 case studies. We provide citations to all sources for each category in the Case Study Sources, mirroring Table 2. Figure 4 compares these results with the distribution of best practices from Security Best Practices.

Design (n=34):

- *Core Agent Design (n=3):* Prompt Engineering (n=2), Multi-Agent (n=0), Other Design Patterns (n=2)
- *Run-Time Controls (n=17):* Monitoring (n=10), Guardrails (n=10), Sanitization (n=4)
- *Human-Agent Interaction (n=11):* Human-in-the-Loop (n=10), Transparency (n=4)
- *Environmental Controls (n=20):* Access Controls (n=17), Sandboxing (n=10)
- *Resource Controls (n=15):* Encryption (n=10), Rate Limiting (n=2), Other Resource Controls (n=6)
- *Defense-in-Depth (n=2)*

Development (n=17):

- *Training Strategies (n=5):* Adversarial Training (n=1), Data Management (n=0), Other Training Strategies (n=4)
- *Evaluation (n=9):* Red-Teaming (n=7), Elicitation (n=1), Metrics (n=2), Other Evaluation Techniques (n=5)
- *Secure System Development (n=8):* Defensive Cybersecurity (n=8), Supply Chain Risk Management (n=4)

Operations (n=16):

- *Deployment (n=14):* Continual Auditing (n=9), Logging (n=9), Incident Response (n=5), Phased Deployment (n=2)
- *Governance (n=11)*, Risk Management (n=9), Organizational Management (n=4), Oversight (n=4), Usage Restrictions (n=6), Legal Considerations (n=0)

Discussion of the Case Studies

Takeaway 8: *Intellectual property concerns hinder transparency.* Figure 4 shows that case studies provide little information about the core agent design and its training strategies, especially in the subcategories of prompt engineering, multi-agent systems, adversarial training, and training data management. These AI components of LLM agents distinguish them from traditional software and thus are crucial pieces of intellectual property. Overall, the case studies highlight the incentives that organizations have to avoid sharing information about security controls closely related to these AI components. Furthermore, the case studies in general provided little information on their security policies, with over 60% of the case studies covering

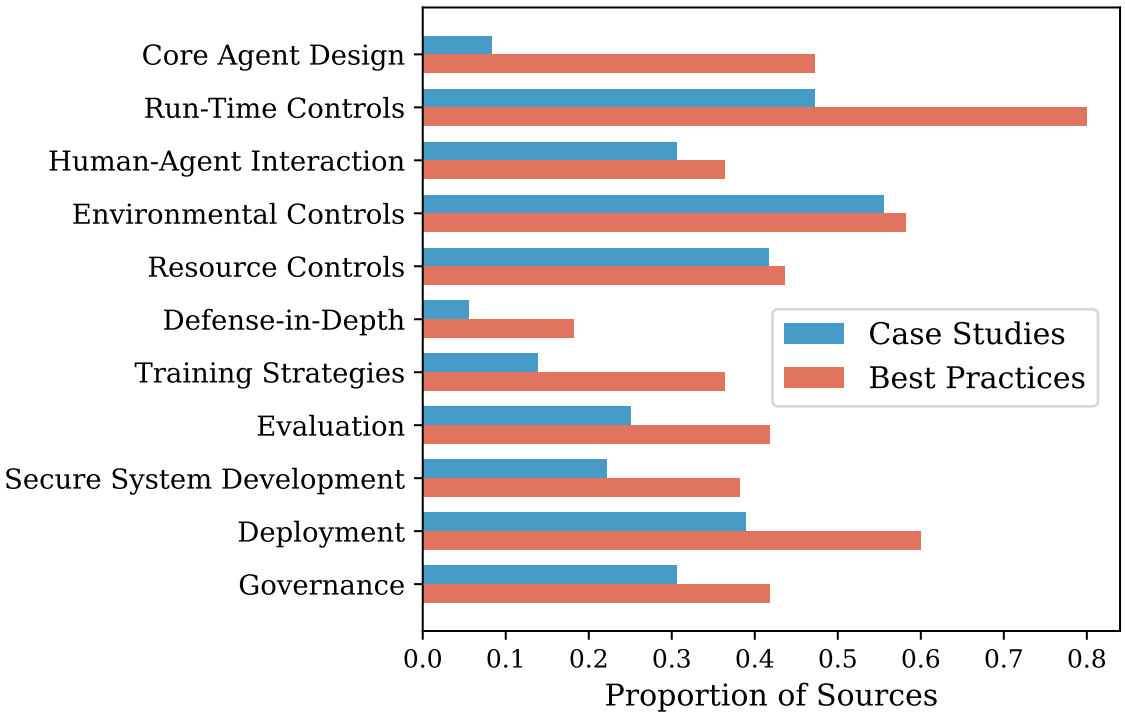

Figure 4: Distribution of security controls across best practice categories. Case studies include security controls implemented in real-world systems, while best practices include recommendations from academia and industry.

fewer than five leaf categories, suggesting a general bias towards secrecy. If security measures cannot be publicly disclosed, they should still be shared with external auditors to ensure accountability [139, 141, 98].

Takeaway 9: *Case studies emphasize user-focused security controls.* The case studies frequently prioritized security measures that protect user privacy and provide users with control over agent behavior. The resource controls implemented in the case studies focused primarily on encrypting and minimizing the retention of user data (e.g., [173, 174]). Environmental controls, the most commonly used security controls, often emphasized protecting and giving users control over their data and other resources (e.g., [175, 176]). Human-agent interaction controls were also frequently used, primarily driven by human-in-the-loop steps that give users the opportunity to approve agent actions (e.g., [177, 178]). This focus on the user highlights the importance of user-centric design in real-world systems, providing an area for academic research and industry guidance to better align with practical considerations.

Takeaway 10: *Public security policies and certifications hold organizations accountable.* Among the four case studies with the broadest coverage of our taxonomy, two came from organizations with public security policies [179, 180] and two from organizations with SOC 2 [181] compliance [182, 183]. This suggests that commitments to public security standards or external audits increase organizational accountability. However, such commitments do not guarantee transparency: several agent providers highlighted their SOC 2 or ISO 27001 compliance while publishing little additional detail about their security controls [184, 185]. Additionally, foundation models [179, 186] and generalist LLM agent platforms [187, 188] also reported comparatively broad sets of security controls, which is particularly important given their role as the basis for various downstream agent applications.

Discussion

We now discuss our overall findings by providing three recommendations, which highlight key themes from our results and offer directions for future work.

R1: Increase Academia–Industry Collaboration.

Our results reveal a complementary approach to LLM agent security between academic and industry work, often centering on differences between research and production or AI and cyber. These differences not only reinforce the utility of our review methodology, but also highlight opportunities for collaboration to better address practical security concerns.

Better engaging researchers in threat intelligence sharing, which is primarily targeted at organizations [46], or providing access to real-world usage data through anonymization techniques like differential privacy or recent LLM-based approaches [189, 190], would better align the two communities.

R2: Develop Agent-Specific Security Controls.

Although our review focused on LLM agent-specific literature, our taxonomies overlap with known LLM threats and best practices [191, 192, 193]. Internal threats, which center on common LLM failure modes like hallucination [194], were the most commonly cited threat surface.

Many of the evaluation recommendations are as relevant to LLMs as to LLM agents. This overlap highlights that LLM agents are, at their core, LLMs—they inherit and amplify LLM threats but also carry forward established best practices. However, it also suggests opportunities for new defenses uniquely tailored to agents.

For example, only two sources [68, 110] cited recent work on agent-specific control flow techniques, which offer security guarantees against prompt injection attacks [195, 196, 197]. Academia, with its focus on emerging threats, is well positioned to advance such approaches.

R3: Expand and Systematize Risk Assessments.

Our contributions provide a framework with which to analyze the security landscape, assess the risk of particular threats, and develop mitigation strategies consisting of consensus-backed security controls. We provide the results of such a process (described in Assessing Selected Threats) for two threats identified in our threat taxonomy.

> *Example 1. Knowledge-base attacks* threaten the data sources from which agents pull information, such as RAG databases or memory systems. These resource-related vulnerabilities map directly to resource controls: encryption and privacy controls mitigate some privacy concerns regarding the retention of potentially sensitive data, and memory controls can purge poisoned memory entries to prevent them from influencing model behavior. Other defenses can protect knowledge bases from external manipulation, including SCRM and usage restrictions. Adversarial training can promote robustness against retrieved poisoned data. Measures such as environmental controls, structured control flows, and rate limiting can contain and minimize losses if agent behavior is hijacked, and run-time controls or HITL steps can block data exfiltration.

Note: Our taxonomy highlights the need for additional research to effectuate this mitigation plan: privacy and memory controls had little emphasis in the literature, despite this being a highly cited vulnerability category. However, data privacy controls are widely used in traditional software [198], indicating a need for

closer alignment between cyber and AI security. The minimal discussion of memory controls highlights an important direction for future work in developing robust memory validation techniques to mitigate memory poisoning attacks.

> *Example 2. Denial-of-Service (DoS) attacks* are a particular attack in the cyber category of resource threats. Defending against DoS requires rate-limiting to bound the number of requests and other resource usage that any individual or group of users can make. Anomalous usage should be monitored and offending accounts should be blocked. Systems should be red-teamed against DoS attacks to ensure that safeguards are adequate, and incident response protocols should respond to in-progress attacks.

Note: DoS attacks are a classic type of cyberattack, and the LLM agent literature provides specific targeted defenses for them, demonstrating the increasing alignment between AI and cybersecurity. However, there are still gaps in the coverage, as monitoring and red-teaming are primarily suggested for agent behavior rather than external cyber threats, and the operational recommendations come primarily from industry sources.

These examples present a first step. The process can be extended to additional threats and deepened by tracing citations further into the literature. Establishing repeatable processes for agent risk analysis should be a priority for both research and practice. **Our work presents such a process**: systematically mapping the threat landscape, taxonomizing security practices, and combining them to yield actionable insights.

Conclusion

LLM agents face a daunting array of threats, posing significant risks to those who build and use them. Yet, with guidance fragmented across sources, developers and organizations lack a coherent path to security.

We conducted a systematic literature review of the LLM agent security landscape to clarify what threats they face, what techniques exist to protect them, and how they are secured in practice. By synthesizing insights from academia, industry, and real-world deployments, our results unify previously scattered insights and provide structure to the current state of LLM agent security.

We contribute a categorization of threats to LLM agents, a taxonomy of actionable security practices, an evaluation of adoption in real-world case studies, and a demonstration of how these components can be combined into repeatable risk assessments.

Securing the future of LLM agents will require a united effort between academia, industry, and government, across AI and cybersecurity domains. We hope this work helps the field move towards standardized, repeatable blueprints for building secure LLM agents.

Appendix

Related Work

Table 3 summarizes the coverage of related academic surveys on LLM security. The table records whether each survey was general or domain-specific (General), whether it followed a systematic review methodology (Sys. Rev.) or was itself a meta-review (Meta-Rev.), and whether it included grey literature (Grey Lit.) or real-world case studies (C.S.). In addition, for each survey we mapped recommended security best practices onto the leaf categories of our best practice taxonomy) to assess coverage (# BP Cat.). Our work is the only survey that addresses all criteria.

Figures 5 and 6 display the number of categories covered by each source for our taxonomy of attack types) and best practices), respectively.

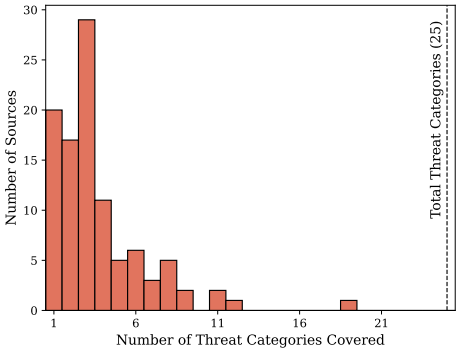

Figure 5: Distribution of the number of threat categories mentioned by each source. No single source contained all 25 identified threat categories.

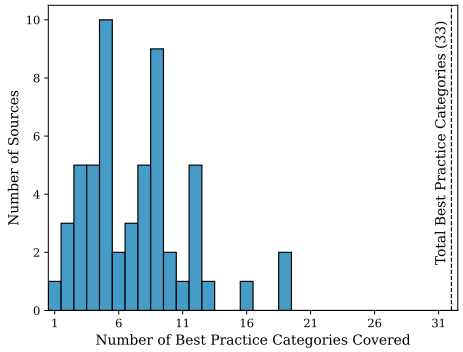

Figure 6: Distribution of the number of best practice categories mentioned by each source. No single source contained all 33 best practice categories.

Table 3: Comparison to related surveys. We evaluate the surveyed academic literature in terms of whether it: was non domain-specific (General), a systematic review (Sys. Rev.), a meta-review (Meta-Rev.), included grey-literature (Grey Lit), or included real-world case-studies (C.S.). We also count the number of best practices from each leaf category in our taxonomy.

Author	General	Sys. Rev.	Meta-Rev.	Grey Lit.	C.S.	# BP Cat.
Campos [139]	✗	✗	✗	✓	✗	4
Deng [20]	✓	✗	✗	✗	✗	12
Li [126]	✗	✗	✗	✗	✗	7
Korbak [40]	✗	✗	✗	✗	✗	2
Schmitz [33]	✗	✗	✗	✗	✗	3
Chen [34]	✗	✓	✗	✗	✗	14
Gan [57]	✓	✗	✗	✗	✓	9
Shanmugarasa [36]	✗	✗	✗	✗	✗	4
Chen [39]	✗	✗	✗	✗	✗	3
Grey [130]	✗	✗	✗	✗	✗	12
Kong [38]	✗	✗	✗	✗	✓	16
Tang [35]	✗	✗	✗	✗	✗	9
Wang [59]	✓	✗	✗	✗	✗	8
He [60]	✓	✗	✗	✗	✗	8
Li [61]	✓	✗	✗	✗	✗	8
Shi [62]	✓	✓	✗	✓	✗	5
Narajala [104]	✓	✓	✗	✓	✓	5
Yan [37]	✗	✓	✗	✗	✗	8
Habler [105]	✗	✗	✗	✗	✗	9
Raza [107]	✓	✓	✗	✗	✓	10
Hammond [19]	✗	✗	✗	✗	✓	7
Ma [68]	✓	✓	✗	✗	✗	9
Shamsujjoha [41]	✗	✓	✗	✗	✗	4
Ko [70]	✗	✗	✗	✗	✗	6
Ours	✓	✓	✓	✓	✓	**36**

Search Strategy Details

Academic Literature: Queries and Constraints

Search Terms: Each of the Semantic Scholar API [48] searches was based on a combination of terms from the following three categories:

- Study scope: *'Survey of'*, *'Review of'*, *'Standards for'*, *'Best practices for'*, *'Open Challenges in'*
- LLM-agent focus: *'LLM Agent'*, *'Agentic AI'*, *'LLM tool use'*, *'Multi agent'*
- Security content: *'Security'*, *'Safety'*, *'Red teaming'*, *'Risks'*, *'Privacy'*

Each query selected one term from each category (e.g., *'Survey of LLM Agent Security'*); our 100 searches covered all unique combinations of terms.

Search Filters: We restricted publication dates to the window July 2020–July 2025 to cover recent work. We filtered results to the 'computer science' category.

Data Collection: We collected the top 15 results per query, sorted by relevance. De-duplication resulted in 478 candidate papers.

Gray Literature: Search Terms and Excluded Domains

Search Terms: Each of the Google search queries was based on a combination of terms from the following three categories:

- Content type: *'Guidelines for'*, *'Best practices for'*, *'White paper for'*, *'Report for'*
- LLM-agent focus: *'AI agent'*, *'Agentic AI'*, *'MCP'*, *'Multi agent'*
- Security focus: *'Security'*, *'Safety'*, *'Risks'*, *'Privacy'*, *'Red Teaming'*, *'Guardrails'*, *'Mitigations'*

Each query selected one term from each category (e.g., *'Guidelines for AI agent Security'*); our 112 searches covered all unique combinations of terms.

Search Filters: Certain websites were excluded to avoid irrelevant content and avoid overlap with the academic search process. These include:

- *Academic*: arxiv.org, neurips.cc, ijcai.org, aaai.org, openreview.net, sciencedirect.com, pmc.ncbi.nlm.nih.gov, ieeexplore.ieee.org, acm.org, nature.com, papers.ssrn.com, researchgate.net, paperswithcode.com
- *Other*: reddit.com, linkedin.com, x.com, github.com

Data Collection: We collected the top 20 results per query. De-duplication resulted in 765 candidate papers.

Case Studies: Google Search Query Template

Search Terms: The following search template was used to investigate the developer website for each entry in the AI Agent Index [29]:

```
site:{developer-domain}
(Agentic OR Agent OR "Model Context Protocol" OR "Tool use")
AND (Secure OR Security OR Safety OR Securing OR Mitigation
     OR Vulnerability OR Threat OR Risk OR Exploit OR Privacy
     OR Cybersecurity OR Authentication OR Authorization
     OR Unauthorized OR Malicious OR Adversary OR Guardrail
     OR Attack OR Red-Team OR "Red Teaming" OR "Red Team"
     OR Privilege OR Escalation)
```

We replaced `{developer-domain}` with each developer website. We conducted searches for each of the 67 entries in the AI Agent Index.

Data Collection: We reviewed up to 50 results per query. Only sources with potential security-related content were retained. 154 candidate sources were collected.

Case Study Sources

In Table 4, we list each case study source which implemented a particular security control in our best practice taxonomy. Some case studies had multiple associated documents; when multiple sources from the case study occurred within a single category we grouped them with parentheses. In Case Studies, we count the number of unique case studies in each category, so grouped citations only count for one occurrence.

Table 4: Case study sources. We cite the case study documents referenced for each best practice category. Documents from the same case study are grouped in parentheses.

Design Development Operations

Category	References
Prompt Engineering	[179, 199]
Multi-Agent	
Other Design Patterns	[200, 199]
Monitoring	[184, 173, 201, 202, 180, 203, 199, 186, 182], ([179, 204])
Guardrails	[200, 178, 205, 186, 206, 176], ([207, 208]), ([179, 204]), ([180, 209]), ([210, 211])
Sanitization	[199, 212, 211, 182]
Human-in-the-Loop	[213, 178, 177, 180, 212, 210, 214, 205, 206, 215]
Transparency	[209, 216, 217, 214]
Access Controls	[173, 215, 218, 219, 214, 217, 220, 212, 221, 180, 222, 204, 178, 183], ([176, 223]), ([224, 175]), ([201, 207, 208])
Sandboxing	[225, 173, 212, 220, 214, 174, 226, 215], ([180, 209]), ([227, 228])
Rate Limiting	[179, 223]
Encryption	[184, 173, 207, 185, 203, 220, 174, 176, 182, 183]
Other Resource Controls	[178, 224, 229, 176, 182, 230]
Defense-in-Depth	[204, 199]
Adversarial Training	[199]
Data Management	
Other Training Strategies	[178, 231, 229, 186]
Red-Teaming	[178, 179, 199, 214, 186, 206], ([180, 209])
Elicitation	[179]
Metrics	[179, 217]
Other Evaluation Techniques	[202, 179, 180, 186], ([217, 210])
Defensive Cybersecurity	[184, 185, 204, 203, 220, 182, 183], ([232, 174])
Supply Chain Risk Management	[204, 220, 182, 183]
Continual Auditing	[180, 224, 203, 220, 176, 182, 183, 233], ([179, 204])
Logging	[184, 173, 207, 204, 180, 212, 214, 234, 183]
Incident Response	[180, 203, 182, 183], ([179, 204])
Phased Deployment	[179, 180]
Risk Management	[185, 203, 220, 176, 182, 183, 235], ([179, 204]), ([180, 209])
Organizational Management	[204, 203, 182, 183]
Oversight	[180, 212, 183], ([176, 223])
Usage Restrictions	[204, 224, 212, 176, 182, 183]
Legal Considerations	

References

[1] T. Kwa, B. West, J. Becker, A. Deng, K. Garcia, M. Hasin, S. Jawhar, M. Kinniment, N. Rush, S. Von Arx, R. Bloom, T. Broadley, H. Du, B. Goodrich, N. Jurkovic, L. H. Miles, S. Nix, T. Lin, N. Parikh, D. Rein, L. J. K. Sato, H. Wijk, D. M. Ziegler, E. Barnes, and L. Chan. "Measuring AI Ability to Complete Long Tasks." *arXiv*. Volume abs/2503.14499. 2025.

[2] D. Ferber, O. S. El Nahhas, G. Wölflein, I. C. Wiest, J. Clusmann, M.-E. Leßmann, S. Foersch, J. Lammert, M. Tschochohei, D. Jäger, M. Salto-Tellez, N. Schultz, D. Truhn, and J. N. Kather. "Development and Validation of an Autonomous Artificial Intelligence Agent for Clinical Decision-Making in Oncology." *Nature Cancer*. Pages 1–13. 2025.

[3] R. Li, X. Wang, D. Berlowitz, J. Mez, H. Lin, and H. Yu. "CARE-AD: A Multi-Agent Large Language Model Framework for Alzheimer's Disease Prediction Using Longitudinal Clinical Notes." *npj Digital Medicine*. Volume 8. Issue 1. Page 541. 2025.

[4] Z. Ren, Y. Zhan, B. Yu, L. Ding, P. Xu, and D. Tao. "Healthcare Agent: Eliciting the Power of Large Language Models for Medical Consultation." *npj Artificial Intelligence*. Volume 1. Issue 1. Page 24. 2025.

[5] X. Shen, L. Wang, Z. Li, Y. Chen, W. Zhao, D. Sun, J. Wang, and W. Ruan. "PenTestAgent: Incorporating LLM Agents Into Automated Penetration Testing." In *Proceedings of the 20th ACM Asia Conference on Computer and Communications Security*. Pages 375–391. 2025.

[6] A. Happe and J. Cito. "Can LLMs Hack Enterprise Networks? Autonomous Assumed Breach Penetration-Testing of Active Directory Networks." *ACM Transactions on Software Engineering and Methodology*. 2025.

[7] Y. Wang, H. Zhai, C. Wang, Q. Hao, N. A. Cohen, R. Foulger, J. A. Handler, and G. Wang. "Can You Walk Me Through It? Explainable SMS Phishing Detection Using LLM-Based Agents." In *Proceedings of the Twenty-First Symposium on Usable Privacy and Security*. Pages 37–56. 2025.

[8] Y. Yu, Z. Yao, H. Li, Z. Deng, Y. Jiang, Y. Cao, Z. Chen, J. Suchow, Z. Cui, R. Liu, Z. Xu, D. Zhang, K. Subbalakshmi, G. Xiong, Y. He, J. Huang, D. Li, and Q. Xie. "FinCon: A Synthesized LLM Multi-Agent System With Conceptual Verbal Reinforcement for Enhanced Financial Decision Making." *Advances in Neural Information Processing Systems*. Volume 37. Pages 137010–137045. 2024.

[9] W. Zhang, L. Zhao, H. Xia, S. Sun, J. Sun, M. Qin, X. Li, Y. Zhao, Y. Zhao, X. Cai, X. Wang, and B. An. "A Multimodal Foundation Agent for Financial Trading: Tool-Augmented, Diversified, and Generalist." In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Pages 4314–4325. 2024.

[10] S. Han, H. Kang, B. Jin, X.-Y. Liu, and S. Y. Yang. "XBRL Agent: Leveraging Large Language Models for Financial Report Analysis." In *Proceedings of the 5th ACM International Conference on AI in Finance*. Pages 856–864. 2024.

[11] X. Ji, L. Zhang, W. Zhang, F. Peng, Y. Mao, X. Liao, and K. Zhang. "LEMAD: LLM-Empowered Multi-Agent System for Anomaly Detection in Power Grid Services." *Electronics*. Volume 14. Issue 15. Page 3008. 2025.

[12] B. Ni, X. Cai, Z. Shen, Z. Meng, J. Zhao, Y. Cheng, and X. Gui. “Intelli-Dispatch-SQL: An LLM-Based Agent for Reliable Text-to-SQL in Power Dispatching.” *Energy and AI*. Page 100591. 2025.

[13] T. Xiao and P. Xu. “Exploring Automated Energy Optimization With Unstructured Building Data: A Multi-Agent Based Framework Leveraging Large Language Models.” *Energy and Buildings*. Volume 322. Page 114691. 2024.

[14] M. Souppaya, K. Scarfone, and D. Dodson. “Secure Software Development Framework (SSDF) Version 1.1.” *NIST Special Publication*. Number 800–218. 2022.

[15] L. Weidinger, J. Uesato, M. Rauh, C. Griffin, P.-S. Huang, J. Mellor, A. Glaese, M. Cheng, B. Balle, A. Kasirzadeh, C. Biles, S. Brown, Z. Kenton, W. Hawkins, T. Stepleton, A. Birhane, L. A. Hendricks, L. Rimell, W. Isaac, J. Haas, S. Legassick, G. Irving, and I. Gabriel. “Taxonomy of Risks Posed by Language Models.” In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. Pages 214–229. 2022.

[16] K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz. “Not What You’ve Signed Up For: Compromising Real-World LLM-Integrated Applications With Indirect Prompt Injection.” In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*. New York, NY, USA: Association for Computing Machinery. Pages 79–90. 2023.

[17] I. Evtimov, A. Zharmagambetov, A. Grattafiori, C. Guo, and K. Chaudhuri. “WASP: Benchmarking Web Agent Security Against Prompt Injection Attacks.” In *ICML 2025 Workshop on Computer Use Agents*. 2025.

[18] O. Peles. “Critical RCE Vulnerability in MCP-Remote: CVE-2025-6514 Threatens LLM Clients.” *JFrog Blog*. 2025. [Online]. Available: https://jfrog.com/blog/2025-6514-critical-mcp-remote-rce-vulnerability/

[19] L. Hammond, A. Chan, J. Clifton, J. Hoelscher-Obermaier, A. Khan, E. McLean, C. Smith, W. Barfuss, J. Foerster, T. Gavenčiak, T. A. Han, E. Hughes, V. Kovařík, J. Kulveit, J. Z. Leibo, C. Oesterheld, C. Schroeder de Witt, N. Shah, M. Wellman, P. Bova, T. Cimpeanu, C. Ezell, Q. Feuillade-Montixi, M. Franklin, E. Kran, I. Krawczuk, M. Lamparth, N. Lauffer, A. Meinke, S. Motwani, A. Reuel, V. Conitzer, M. Dennis, I. Gabriel, A. Gleave, G. Hadfield, N. Haghtalab, A. Kasirzadeh, S. Krier, K. Larson, J. Lehman, D. C. Parkes, G. Piliouras, and I. Rahwan. “Multi-Agent Risks from Advanced AI.” *arXiv*. Volume abs/2502.14143. February 2025.

[20] Z. Deng, Y. Guo, C. Han, W. Ma, J. Xiong, S. Wen, and Y. Xiang. “AI Agents Under Threat: A Survey of Key Security Challenges and Future Pathways.” *ACM Computing Surveys*. Volume 57. Issue 7. February 2025.

[21] A. Chuvakin. “Cloud CISO Perspectives: How Google Secures AI Agents.” *Google Cloud Blog*. 2025. [Online]. Available: https://cloud.google.com/blog/products/identity-security/cloud-ciso-perspectives-how-google-secures-ai-agents

[22] Anthropic. “Activating AI Safety Level 3 Protections.” *Anthropic News*. 2025. [Online]. Available: https://www.anthropic.com/news/activating-asl3-protections

[23] U.S. AI Safety Institute Technical Staff. “Technical Blog: Strengthening AI Agent Hijacking Evaluations.” *NIST*. January 2025. [Online]. Available: https://www.nist.gov/news-events/news/2025/01/technical-blog-strengthening-ai-agent-hijacking-evaluations

[24] U.S. Government Accountability Office. “Science & Tech Spotlight: AI Agents — U.S. GAO.” *U.S. GAO*. 2025.

[25] N. H. Conradie and S. K. Nagel. “No Agent in the Machine: Being Trustworthy and Responsible About AI.” *Philosophy & Technology*. Volume 37. Number 2. Page 72. 2024.

[26] B. Bent. “The Term ‘Agent’ Has Been Diluted Beyond Utility and Requires Redefinition.” *arXiv*. Volume abs/2508.05338. 2025.

[27] E. Perrier and M. T. Bennett. “Position: Stop Acting Like Language Model Agents Are Normal Agents.” *arXiv*. Volume abs/2502.10420. 2025.

[28] A. Chan, R. Salganik, A. Markelius, C. Pang, N. Rajkumar, D. Krasheninnikov, L. Langosco, Z. He, Y. Duan, M. Carroll, M. Lin, A. Mayhew, K. Collins, M. Molamohammadi, J. Burden, W. Zhao, S. Rismani, K. Voudouris, U. Bhatt, A. Weller, D. Krueger, and T. Maharaj. “Harms from Increasingly Agentic Algorithmic Systems.” In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*. Pages 651–666. 2023.

[29] S. Casper, L. Bailey, R. Hunter, C. Ezell, E. Cabalé, M. Gerovitch, S. Slocum, K. Wei, N. Jurkovic, A. Khan, P. Christoffersen, A. P. Ozisik, R. Trivedi, D. Hadfield-Menell, and N. Kolt. “The AI Agent Index.” *arXiv*. Volume abs/2502.01635. 2025.

[30] K. Chen, M. Cusumano-Towner, B. Huval, A. Petrenko, J. Hamburger, V. Koltun, and P. Krähnbühl. “Reinforcement Learning for Long-Horizon Interactive LLM Agents.” *arXiv*. Volume abs/2502.01600. 2025.

[31] A. Sinha, K. Grimes, J. Lucassen, M. Feffer, N. VanHoudnos, Z. S. Wu, and H. Heidari. “From Firewalls to Frontiers: AI Red-Teaming Is a Domain-Specific Evolution of Cyber Red-Teaming.” *arXiv*. Volume abs/2509.11398. 2025.

[32] CNSS. “Committee on National Security Systems (CNSS) Glossary.” *Committee on National Security Systems*. Technical Report. 2015.

[33] C. Schmitz, J. Rystrøm, and J. Batzner. “Oversight Structures for Agentic AI in Public-Sector Organizations.” In *Proceedings of the 1st Workshop for Research on Agent Language Models*. E. Kamalloo, N. Gontier, X. H. Lu, N. Dziri, S. Murty, and A. Lacoste, Editors. Vienna, Austria: Association for Computational Linguistics. Pages 298–308. July 2025.

[34] A. Chen, Y. Wu, J. Zhang, S. Yang, J.-T. Huang, K. Wang, W. Wang, and S. Wang. “A Survey on the Safety and Security Threats of Computer-Using Agents: Jarvis or Ultron?” *arXiv*. Volume abs/2505.10924. 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:278715451

[35] X. Tang, Q. Jin, K. Zhu, T. Yuan, Y. Zhang, W. Zhou, M. Qu, Y. Zhao, J. Tang, Z. Zhang, A. Cohan, Z. Lu, and M. B. Gerstein. “Risks of AI Scientists: Prioritizing Safeguarding over Autonomy.” *Nature Communications*. Volume 16. 2025.

[36] Y. Shanmugarasa, M. Ding, C. M. Arachchige, and T. Rakotoarivelo. “SoK: The Privacy Paradox of Large Language Models: Advancements, Privacy Risks, and Mitigation.” In *Proceedings of the 20th ACM Asia Conference on Computer and Communications Security*. Pages 425–441. 2025.

[37] B. Yan, K. Li, M. Xu, Y. Dong, Y. Zhang, Z. Ren, and X. Cheng. "On Protecting the Data Privacy of Large Language Models (LLMs) and LLM Agents: A Literature Review." *High-Confidence Computing*. Volume 5. Issue 2. Page 100300. 2025.

[38] D. Kong, S. Lin, Z. Xu, Z. Wang, M. Li, Y. Li, Y. Zhang, H. Peng, Z. Sha, Y. Li, C. Lin, X. Wang, X. Liu, N. Zhang, C.-J. Chen, M. K. Khan, and M. Han. "A Survey of LLM-Driven AI Agent Communication: Protocols, Security Risks, and Defense Countermeasures." *arXiv*. Volume abs/2506.19676. 2025.

[39] C. Chen, Z. Zhang, I. Khalilov, B. Guo, S. A. Gebreegziabher, Y. Ye, Z. Xiao, Y. Yao, T. Li, and T. J.-J. Li. "Toward a Human-Centered Evaluation Framework for Trustworthy LLM-Powered GUI Agents." *arXiv*. Volume abs/2504.17934. 2025.

[40] T. Korbak, M. Balesni, B. Shlegeris, and G. Irving. "How to Evaluate Control Measures for LLM Agents? A Trajectory from Today to Superintelligence." *arXiv*. Volume abs/2504.05259. 2025.

[41] M. Shamsujjoha, Q. Lu, D. Zhao, and L. Zhu. "Swiss Cheese Model for AI Safety: A Taxonomy and Reference Architecture for Multi-Layered Guardrails of Foundation Model-Based Agents." *2025 IEEE 22nd International Conference on Software Architecture*. Pages 37–48. 2025.

[42] K. Huang. "Agentic AI Threat Modeling Framework: MAESTRO." *Cloud Security Alliance Blog*. February 2025. [Online]. Available: https://cloudsecurityalliance.org/blog/2025/02/06/agentic-ai-threat-modeling-framework-maestro

[43] A. Dragan, R. Shah, F. Flynn, and S. Legg. "Taking a Responsible Path to AGI." *Google DeepMind Blog*. April 2025. [Online]. Available: https://deepmind.google/discover/blog/taking-a-responsible-path-to-agi/

[44] IBM. "IBM Whitepaper: Accountability and Risk Matter in Agentic AI." 2025.

[45] A. Sinha, J. Lucassen, K. Grimes, M. Feffer, M. Soto, H. Heidari, and N. VanHoudnos. "What Can Generative AI Red-Teaming Learn from Cyber Red-Teaming?" *Software Engineering Institute, Technical Report CMU/SEI-2025-TR-006*. 2025. Available: doi.org/10.1184/R1/29410136

[46] C. Johnson. "Guide to Cyber Threat Information Sharing." *NIST Special Publication*. Number 800-150. 2016.

[47] A. P. Siddaway, A. M. Wood, and L. V. Hedges. "How to Do a Systematic Review: A Best Practice Guide for Conducting and Reporting Narrative Reviews, Meta-Analyses, and Meta-Syntheses." *Annual Review of Psychology*. Volume 70. Number 1. Pages 747–770. 2019.

[48] R. M. Kinney, C. Anastasiades, R. Authur, I. Beltagy, J. Bragg, A. Buraczynski, I. Cachola, S. Candra, Y. Chandrasekhar, A. Cohan, M. Crawford, D. Downey, J. Dunkelberger, O. Etzioni, R. Evans, S. Feldman, J. Gorney, D. W. Graham, F. Hu, R. Huff, D. King, S. Kohlmeier, B. Kuehl, M. Langan, D. Lin, H. Liu, K. Lo, J. Lochner, K. MacMillan, T. C. Murray, C. Newell, S. R. Rao, S. Rohatgi, P. Sayre, Z. Shen, A. Singh, L. Soldaini, S. Subramanian, A. Tanaka, A. D. Wade, L. M. Wagner, L. L. Wang, C. Wilhelm, C. Wu, J. Yang, A. Zamarron, M. van Zuylen, and D. S. Weld. "The Semantic Scholar Open Data Platform." *arXiv*. Volume abs/2301.10140. 2023.

[49] V. Garousi, M. Felderer, and M. V. Mäntylä. "Guidelines for Including Grey Literature and Conducting Multivocal Literature Reviews in Software Engineering." *Information and Software Technology*. Volume 106. Pages 101–121. 2019.

[50] K. Godin, J. Stapleton, S. I. Kirkpatrick, R. M. Hanning, and S. T. Leatherdale. "Applying Systematic Review Search Methods to the Grey Literature: A Case Study Examining Guidelines for School-Based Breakfast Programs in Canada." *Systematic Reviews*. Volume 4. Number 1. Page 138. 2015.

[51] NCSC and CISA. "Guidelines for Secure AI System Development." NCSC, Tech. Rep., 2023.

[52] Z. Chen, J. Chen, J. Chen, and M. Sra. "Standard Benchmarks Fail—Auditing LLM Agents in Finance Must Prioritize Risk." *arXiv*. Volume abs/2502.15865. 2025.

[53] W. Xu, C. Huang, S. Gao, and S. Shang. "LLM-based Agents for Tool Learning: A Survey." *Data Science and Engineering*. 2025.

[54] E. Shayegani, M. A. A. Mamun, Y. Fu, P. Zaree, Y. Dong, and N. B. Abu-Ghazaleh. "Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks." *arXiv*. Volume abs/2310.10844, 2023.

[55] G. Molinari and F. Ciravegna. "Towards Pervasive Distributed Agentic Generative AI—A State of the Art." *arXiv*. Volume abs/2506.13324. 2025.

[56] J. Fang, Z. Yao, R. Wang, H. Ma, X. Wang, and T.-S. Chua. "We Should Identify and Mitigate Third-Party Safety Risks in MCP-Powered Agent Systems." *arXiv*. Volume abs/2506.13666. 2025.

[57] Y. Gan, Y. Yang, Z. Ma, P. He, R. Zeng, Y. Wang, Q. Li, C. Zhou, S. Li, T. Wang, Y. Gao, Y. Wu, and S. Ji. "Navigating the Risks: A Survey of Security, Privacy, and Ethics Threats in LLM-based Agents." *arXiv*. Volume abs/2411.09523. 2024.

[58] S. Wang, T. Zhu, B. Liu, M. Ding, X. Guo, D. Ye, W. Zhou, and P. S. Yu. "Unique Security and Privacy Threats of Large Language Models: A Comprehensive Survey." *arXiv*. Volume abs/2406.07973. 2024.

[59] K. Wang, G. Zhang, Z. Zhou, J. Wu, M. Yu, S. Zhao, C. Yin, J. Fu, Y. Yan, H. Luo, L. Lin, Z. Xu, H. Lu, X. Cao, X. Zhou, W. Jin, F. Meng, J. Mao, H. Wu, M. Wang, F. Zhang, J. Fang, C. Liu, Y. Zhang, Q. Li, C. Guo, Y. Qin, Y. Ding, D. Hong, J. Ji, X. Li, Y. Jiang, D. Wang, Y. Huang, Y. Guo, J. tse Huang, Y. Yue, W.-S. Huang, G. Wan, T.-C. Li, L. Bai, J. Zhang, Q. Guo, J. Wang, T. Chen, J. T. Zhou, X. Jia, W. Sun, C. Wu, J. Chen, X. Hu, Y. Li, X. Wang, N. Zhang, A. T. Luu, G. Xu, T. Zhang, X. mei Ma, X. Wang, B. An, J. Sun, M. Bansal, S. Pan, Y. Elovici, B. Kailkhura, B. Li, Y.-G. Yang, H. Li, W. Xu, Y. Sun, W. Wang, Q. Li, K. Tang, Y. Jiang, F. Juefei-Xu, H. Xiong, X. Wang, S. Yan, D. Tao, P. S. Yu, Q.-P. Wen, and Y. Liu. "A Comprehensive Survey in LLM(-Agent) Full Stack Safety: Data, Training and Deployment." *arXiv*. Volume abs/2504.15585. 2025.

[60] F. He, T. Zhu, D. Ye, B. Liu, W. Zhou, and P. S. Yu. "The Emerged Security and Privacy of LLM Agents: A Survey with Case Studies." *arXiv*. Volume abs/2407.19354. 2024.

[61] M. Q. Li and B. C. M. Fung. "Security Concerns for Large Language Models: A Survey." *arXiv*. Volume abs/2505.18889. 2025.

[62] D. Shi, T. Shen, Y. Huang, Z. Li, Y. Leng, R. Jin, C. Liu, X. Wu, Z. Guo, L. Yu, L. Shi, B. Jiang, and D. Xiong. "Large Language Model Safety: A Holistic Survey." *arXiv*. Volume abs/2412.17686. 2024.

[63] S. Chern, Z. Fan, and A. Liu. "Combating Adversarial Attacks with Multi-Agent Debate." *arXiv*. Volume abs/2401.05998. 2024.

[64] T. Raheja and N. Pochhi. “Recent Advancements in LLM Red-Teaming: Techniques, Defenses, and Ethical Considerations.” *arXiv*. Volume abs/2410.09097. 2024.

[65] S. Naihin, D. Atkinson, M. Green, M. Hamadi, C. Swift, D. Schonholtz, A. T. Kalai, and D. Bau. “Testing Language Model Agents Safely in the Wild.” *arXiv*. Volume abs/2311.10538, 2023.

[66] W. Xu and K. K. Parhi. “A Survey of Attacks on Large Language Models.” *arXiv*. Volume abs/2505.12567. 2025.

[67] O. Delaney, O. Guest, and Z. Williams. “Mapping Technical Safety Research at AI Companies: A Literature Review and Incentives Analysis.” *arXiv*. Volume abs/2409.07878. 2024.

[68] X. Ma, Y. Gao, Y. Wang, R. Wang, X. Wang, Y. Sun, Y. Ding, H. Xu, Y. Chen, Y. Zhao, H. Huang, Y. Li, J. Zhang, X. Zheng, Y. Bai, H. Ding, Z. Wu, X. Qiu, J. Zhang, Y. Li, J. Sun, C. Wang, J. Gu, B. Wu, S. Chen, T. Zhang, Y. Liu, M. Gong, T. Liu, S. Pan, C. Xie, T. Pang, Y. Dong, R. Jia, Y. Zhang, S.-J. Ma, X. Zhang, N. Gong, C. Xiao, S. Erfani, B. Li, M. Sugiyama, D. Tao, J. Bailey, and Y.-G. Jiang. “Safety at Scale: A Comprehensive Survey of Large Model Safety.” *arXiv*. Volume abs/2502.05206. 2025.

[69] H. Song, Y. Shen, W. Luo, L. Guo, T. Chen, J. Wang, B. Li, X. Zhang, and J. Chen. “Beyond the Protocol: Unveiling Attack Vectors in the Model Context Protocol Ecosystem.” *arXiv*. Volume abs/2506.02040. 2025.

[70] R. Ko, J. Jeong, S. Zheng, C. Xiao, T. Kim, M. Onizuka, and W. Shin. “Seven Security Challenges That Must Be Solved in Cross-Domain Multi-Agent LLM Systems.” *arXiv*. Volume abs/2505.23847. 2025.

[71] Google. “Responsible AI: Our 2024 Report and Ongoing Work.” *Google Blog*. February 2025. [Online]. Available: https://blog.google/technology/ai/responsible-ai-2024-report-ongoing-work/

[72] Microsoft. “Secure Azure Platform Services (PaaS) for AI – Cloud Adoption Framework.” *Microsoft Learn*. 2025. [Online]. Available: https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/scenarios/ai/platform/security

[73] J. C. Lu and Royce. “AI Agents Are Here. So Are the Threats.” *Palo Alto Networks Unit 42 Blog*. May 2025. [Online]. Available: https://unit42.paloaltonetworks.com/agentic-ai-threats/

[74] D. Song. “Towards Building Safe and Trustworthy AI Agents and a Path for Science- and Evidence-Based AI Policy.” *Berkeley RDI*. 2025. [Online]. Available: https://rdi.berkeley.edu/llm-agents/assets/dawn-agent-safety.pdf

[75] D. Kumar. “Agent Red-Teaming: Exposing Vulnerabilities in Autonomous Financial AI Systems.” *Enkrypt AI Blog*. 2025. [Online]. Available: https://www.enkryptai.com/blog/agent-red-teaming-exposing-vulnerabilities-in-autonomous-financial-ai-systems

[76] M. Ali. “Model Context Protocol (MCP) Security.” *Writer Engineering Blog*. 2025. [Online]. Available: https://writer.com/engineering/mcp-security-considerations/

[77] A. Bodhankar. “How to Safeguard AI Agents for Customer Service with NVIDIA NeMo Guardrails.” *NVIDIA Developer Blog*. January 2025. [Online]. Available: https://developer.nvidia.com/blog/how-to-safeguard-ai-agents-for-customer-service-with-nvidia-nemo-guardrails/

[78] Enkrypt AI. “Ensuring AI Safety and Compliance: Comparative Study of LLM Guardrails.” *Enkrypt AI*. 2025. [Online]. Available: https://www.enkryptai.com/blog/ensuring-ai-safety-and-compliance-comparative-study-of-llm-guardrails

[79] D. Gilmore. “Securing AI Agent Authentication: Risks and Best Practices.” *Threat Intelligence Blog*. January 2025. [Online]. Available: https://www.threatintelligence.com/a-new-era-of-agentware-malicious-ai-agents-as-emerging-threat-vectors

[80] R. Golabek. “Red Teaming LLMs to AI Agents: Beyond One-Shot Prompts.” *MyKubert Blog*. September 2024. [Online]. Available: https://mykubert.com/blog/red-teaming-llms-ai-agents-threats/

[81] D. Nissani, B. Risher, L. Ross, and S. Tan. “Automating the Adversary: Designing a Scalable Framework for Red Teaming AI.” *Salesforce Blog*. November 2024. [Online]. Available: https://www.salesforce.com/blog/automated-framework-for-red-teaming-ai/

[82] K. Park. “MCP PAM as the Next Step Beyond Guardrails.” *QueryPie White Paper*. 2025. [Online]. Available: https://www.querypie.com/resources/discover/white-paper/16/next-step-mcp-pam

[83] I. Barberá. “AI Privacy Risks and Mitigations for Large Language Models (LLMs).” *European Data Protection Board*. 2025. [Online]. Available: https://www.edpb.europa.eu/our-work-tools/our-documents/support-pool-experts-projects/ai-privacy-risks-mitigations-large_en

[84] Amazon Web Services. “Architecting Secure MCP Solutions on AWS: From Threats to Mitigations.” *AWS Builder Center*. 2025. [Online]. Available: https://builder.aws.com/content/2vmTtkYI0FIIqT1NZHj6SZ40tnU/architecting-secure-mcp-solutions-on-aws-from-threats-to-mitigations

[85] D. Berrick. “Minding Mindful Machines: AI Agents and Data Protection Considerations.” *Future of Privacy Forum*. 2025. [Online]. Available: https://fpf.org/blog/minding-mindful-machines-ai-agents-and-data-protection-considerations/

[86] H. Toner, J. Bansemer, K. Crichton, M. Burtell, T. Woodside, A. Lior, A. J. Lohn, A. Acharya, B. Cibralic, C. Painter, C. O’Keefe, I. Gabriel, K. Fisher, K. Ramakrishnan, K. Jackson, N. Kolt, R. Crootof, and S. Chatterjee. “Through the Chat Window and Into the Real World: Preparing for AI Agents.” *Georgetown University CSET Report*. 2025.

[87] S. Díaz, C. Kern, and K. Olive. “Google’s Approach for Secure AI Agents.” *Google Research*. 2025. [Online]. Available: https://research.google/pubs/an-introduction-to-googles-approach-for-secure-ai-agents/

[88] T. Olavsrud. “NVIDIA Intros New Guardrail Microservices for Agentic AI.” *CIO Magazine*. 2025. [Online]. Available: https://www.cio.com/article/3803583/nvidia-intros-new-guardrail-microservices-for-agentic-ai.html

[89] J. Selvi. “5 MCP Security Tips.” *NCC Group*. 2025. [Online]. Available: https://www.nccgroup.com/

[90] C. Bronsdon. “Threat Modeling for Multi-Agent AI: How to Identify and Prevent Systemic Risks.” *Galileo Blog*. 2025. [Online]. Available: https://galileo.ai/blog/threat-modeling-multi-agent-ai

[91] K. Huang. “Agentic AI Threat Modeling Framework: MAESTRO.” *Cloud Security Alliance*. February 2025. [Online]. Available: https://cloudsecurityalliance.org/blog/2025/02/06/agentic-ai-threat-modeling-framework-maestro

[92] Amazon Web Services. “Navigating the Security Landscape of Generative AI.” *AWS Whitepaper*. 2025. [Online]. Available: https://docs.aws.amazon.com/whitepapers/latest/navigating-security-landscape-genai/navigating-security-landscape-genai.html

[93] T. Savant. “Guide to Ethical Red Teaming: Prompt Injection Attacks on Multi-Modal LLM Agents.” *Test Savant*. March 2025. [Online]. Available: https://testsavant.ai/red-teaming-guide/

[94] T. Elias. “Agentic AI Security: OWASP Threats and How to Defend Against Them.” *Human Security Blog*. April 2025. [Online]. Available: https://www.humansecurity.com/learn/blog/agentic-ai-security-owasp-threats/

[95] E. Sherman, S. Shattuck, N. Singh, and I. Eisenberg. “From Assistant to Agent: Governing Autonomous AI.” *Credo AI*. 2025. [Online]. Available: https://www.credo.ai/recourseslongform/from-assistant-to-agent-navigating-the-governance-challenges-of-increasingly-autonomous-ai

[96] Anthropic. “Recommendations for Technical AI Safety Research Directions.” *Anthropic Alignment*. 2025. [Online]. Available: https://alignment.anthropic.com/2025/recommended-directions/

[97] F. C. Gabarda. “Model Context Protocol (MCP): Understanding Security Risks and Controls.” *Red Hat Blog*. 2025. [Online]. Available: https://www.redhat.com/en/blog/model-context-protocol-mcp-understanding-security-risks-and-controls

[98] H. Wong and T. Saade. “Report Warns of Agentic AI Cyber Risks.” *R Street Institute*. 2025. [Online]. Available: https://www.rstreet.org/commentary/report-warns-of-agentic-ai-cyber-risks/

[99] Teleport. “Secure MCP: Securing How AI Interacts with Your Data Sources.” *Teleport*. 2025. [Online]. Available: https://goteleport.com/use-cases/secure-model-context-protocol/

[100] OpenAI. “Our Updated Preparedness Framework.” *OpenAI*. 2025. [Online]. Available: https://openai.com/index/updating-our-preparedness-framework/

[101] Stytch. “AI Agent Security: How to Authenticate, Authorize, and Monitor Agents.” *Stytch Blog*. 2025. [Online]. Available: https://stytch.com/blogs/ai-agent-security-explained/

[102] Microsoft. “Responsible AI: Ethical Policies and Practices.” *Microsoft AI*. 2025. [Online]. Available: https://www.microsoft.com/en-us/ai/responsible-ai

[103] Q. Li and Y. Xie. “From Glue-Code to Protocols: A Critical Analysis of A2A and MCP Integration for Scalable Agent Systems.” *arXiv*. Volume abs/2505.03864. 2025.

[104] V. S. Narajala and O. Narayan. “Securing Agentic AI: A Comprehensive Threat Model and Mitigation Framework for Generative AI Agents.” *arXiv*. Volume abs/2504.19956. 2025.

[105] I. Habler, K. Huang, V. S. Narajala, and P. Kulkarni. “Building a Secure Agentic AI Application Leveraging A2A Protocol.” *arXiv*. Volume abs/2504.16902. 2025.

[106] P. He, Y. Lin, S. Dong, H. Xu, Y. Xing, and H. Liu. “Red-Teaming LLM Multi-Agent Systems via Communication Attacks.” *arXiv*. Volume abs/2502.14847. 2025.

[107] S. Raza, R. Sapkota, M. Karkee, and C. Emmanouilidis. “TRiSM for Agentic AI: A Review of Trust, Risk, and Security Management in LLM-Based Agentic Multi-Agent Systems.” *arXiv*. Volume abs/2506.04133. 2025.

[108] M. A. Ferrag, N. Tihanyi, and M. Debbah. “From LLM Reasoning to Autonomous AI Agents: A Comprehensive Review.” *arXiv*. Volume abs/2504.19678. 2025.

[109] J. Sotiropoulos. “Securing AI’s New Frontier: The Power of Open Collaboration on MCP Security.” *OWASP GenAI Security Project Blog*. April 2025. [Online.] Available: https://genai.owasp.org/2025/04/22/securing-ais-new-frontier-the-power-of-open-collaboration-on-mcp-security/

[110] R. S. S. Kumar. “New Whitepaper Outlines the Taxonomy of Failure Modes in AI Agents.” *Microsoft Security Blog*. April 2025. [Online]. Available: https://www.microsoft.com/en-us/security/blog/2025/04/24/new-whitepaper-outlines-the-taxonomy-of-failure-modes-in-ai-agents/

[111] R. McCarthy. “MCP and LLM Security Research Briefing.” *Wiz Blog*. 2025. [Online]. Available: https://www.wiz.io/blog/mcp-security-research-briefing

[112] CSA. “Agentic AI Red Teaming Guide.” *Cloud Security Alliance*. 2025. [Online]. Available: https://cloudsecurityalliance.org/artifacts/agentic-ai-red-teaming-guide

[113] D. Weston. “Securing the Model Context Protocol: Building a Safer Agentic Future on Windows.” *Microsoft Blog*. May 2025. [Online]. Available: https://blogs.windows.com/windowsexperience/2025/05/19/securing-the-model-context-protocol-building-a-safer-agentic-future-on-windows/

[114] IBM. “Scaling Responsible Agentic AI.” *IBM Think Insights*. April 2025. [Online]. Available: https://www.ibm.com/think/insights/scale-responsible-agentic-ai

[115] OWASP. “Multi-Agentic System Threat Modeling Guide v1.0.” *OWASP*. 2025. [Online]. Available: https://genai.owasp.org/resource/multi-agentic-system-threat-modeling-guide-v1-0/

[116] S. Young. “Protecting Against Indirect Prompt Injection Attacks in MCP.” *Microsoft Developer Blog*. April 2025. [Online]. Available: https://developer.microsoft.com/blog/protecting-against-indirect-injection-attacks-mcp

[117] C. Posa. “Deep Dive: MCP and A2A Attack Vectors for AI Agents.” *Solo.io Blog*. 2025. [Online]. Available: https://www.solo.io/blog/deep-dive-mcp-and-a2a-attack-vectors-for-ai-agents

[118] Model Context Protocol. “Security Best Practices.” *Model Context Protocol Specification*. 2025. [Online]. Available: https://modelcontextprotocol.io/specification/draft/basic/security_best_practices

[119] K. Schulz, J. Martin, M. Kan, K. Yeung, C. McCauley, and L. Ring. “MCP: Model Context Pitfalls in an Agentic World.” *HiddenLayer Innovation Hub*. 2025. [Online]. Available: https://hiddenlayer.com/innovation-hub/mcp-model-context-pitfalls-in-an-agentic-world/

[120] L. Beurer-Kellner and M. Fischer. “MCP Security Notification: Tool Poisoning Attacks.” *Invariant Labs*. April 2025. [Online]. Available: https://invariantlabs.ai/blog/mcp-security-notification-tool-poisoning-attacks

[121] I. Beber. “EscapeRoute: Breaking the Scope of Anthropic’s Filesystem MCP Server.” *Cymulate Blog*. 2025. [Online]. Available: https://cymulate.com/blog/cve-2025-53109-53110-escaperoute-anthropic/

[122] G. Manor. “Human-in-the-Loop for AI Agents: Best Practices, Frameworks, Use Cases, and Demo.” *Permit.io Blog*. June 2025. [Online]. Available: https://www.permit.io/blog/human-in-the-loop-for-ai-agents-best-practices-frameworks-use-cases-and-demo

[123] C. Bronsdon. “Detect and Prevent Malicious Agents in Multi-Agent Systems.” *Galileo Blog*. 2025. [Online]. Available: https://galileo.ai/blog/multi-agent-systems-exploits

[124] I. Alvas. “3 Takeaways from the OWASP Agentic AI Security Research.” *Entro Security Blog*. 2025. [Online]. Available: https://entro.security/blog/agentic-ai-owasp-research/

[125] A. Sarkar and S. Sarkar. “Survey of LLM Agent Communication with MCP: A Software Design Pattern Centric Review.” *arXiv*. Volume abs/2506.05364. 2025.

[126] Y. Li, H. Wen, W. Wang, X. Li, Y. Yuan, G. Liu, J. Liu, W. Xu, X. Wang, Y. Sun, R. Kong, Y. Wang, H. Geng, J. Luan, X. Jin, Z.-L. Ye, G. Xiong, F. Zhang, X. Li, M. Xu, Z. Li, P. Li, Y. Liu, Y. Zhang, and Y. Liu. “Personal LLM Agents: Insights and Survey About the Capability, Efficiency, and Security.” *arXiv*. Volume abs/2401.05459. 2024.

[127] B. Xia, Q. Lu, L. Zhu, Z. Xing, D. Zhao, and H. Zhang. “Evaluation-Driven Development of LLM Agents: A Process Model and Reference Architecture.” *arXiv*. Volume abs/2411.13768. 2024.

[128] D.-M. Córdova-Esparza. “AI-Powered Educational Agents: Opportunities, Innovations, and Ethical Challenges.” *Information*. Volume 16. Number 6. Page 469. 2025. Available: https://doi.org/10.3390/info16060469

[129] P. Rouzrokh, B. Khosravi, S. Faghani, M. Moassefi, M. Shariatnia, P. Rouzrokh, and B. J. Erickson. “A Current Review of Generative AI in Medicine: Core Concepts, Applications, and Current Limitations.” *Current Reviews in Musculoskeletal Medicine*. 2025.

[130] M. Grey and C.-R. Ségerie. “Safety by Measurement: A Systematic Literature Review of AI Safety Evaluation Methods.” *arXiv*. Volume abs/2505.05541. 2025.

[131] C. S. de Witt. “Open Challenges in Multi-Agent Security: Towards Secure Systems of Interacting AI Agents.” *arXiv*. Volume abs/2505.02077. 2025.

[132] Y. Wang, D. Xue, S. Zhang, and S. Qian. “BadAgent: Inserting and Activating Backdoor Attacks in LLM Agents.” In *Annual Meeting of the Association for Computational Linguistics*. 2024.

[133] W. Zeng, H. Zhu, C. Qin, H. Wu, Y. Cheng, S. Zhang, X. Jin, Y. Shen, Z. Wang, F. Zhong, and H. Xiong. “Multi-Level Value Alignment in Agentic AI Systems: Survey and Perspectives.” *arXiv*. Volume abs/2506.09656. 2025.

[134] Salesforce. “Define the Agent Guardrails.” *Salesforce Trailhead*. 2025. [Online]. Available: https://trailhead.salesforce.com/content/learn/modules/agentforce-agent-planning/define-the-agent-guardrails

[135] D. McDaniel. “GitHub Copilot Security and Privacy Concerns: Understanding the Risks and Best Practices.” *GitGuardian Blog*. March 2025. [Online]. Available: https://blog.gitguardian.com/github-copilot-security-and-privacy/

[136] Salesforce. “Salesforce AI Research Delivers New Benchmarks, Guardrails, and Models to Make Future Agents More Intelligent, Trusted, and Versatile.” *Salesforce News*. May 2025. [Online]. Available: https://www.salesforce.com/news/stories/ai-research-agentic-advancements/

[137] B. Baker, J. Huizinga, A. Madry, W. Zaremba, J. Pachocki, and D. Farhi. “Detecting Misbehavior in Frontier Reasoning Models.” *OpenAI*. 2025. [Online]. Available: https://openai.com/index/chain-of-thought-monitoring/

[138] Anthropic. “Agentic Misalignment: How LLMs Could Be Insider Threats.” *Anthropic Research*. 2025. [Online]. Available: https://www.anthropic.com/research/agentic-misalignment

[139] S. Campos, H. Papadatos, F. Roger, C. Touzet, O. Quarks, and M. Murray. “A Frontier AI Risk Management Framework: Bridging the Gap Between Current AI Practices and Established Risk Management.” *arXiv*. Volume abs/2502.06656. 2025.

[140] R. Pankajakshan, S. Biswal, Y. Govindarajulu, and G. Gressel. “Mapping LLM Security Landscapes: A Comprehensive Stakeholder Risk Assessment Proposal.” *arXiv*. Volume abs/2403.13309. 2024.

[141] Anthropic. “Anthropic’s Responsible Scaling Policy.” *Anthropic News*. 2025. [Online]. Available: https://www.anthropic.com/news/anthropics-responsible-scaling-policy

[142] B. Bullwinkel and R. S. S. Kumar. “3 Takeaways from Red Teaming 100 Generative AI Products.” *Microsoft Security Blog*. January 2025. [Online]. Available: https://www.microsoft.com/en-us/security/blog/2025/01/13/3-takeaways-from-red-teaming-100-generative-ai-products/

[143] OWASP. “Agentic AI: Threats and Mitigations.” *OWASP*. February 2024. [Online]. Available: https://genai.owasp.org/resource/agentic-ai-threats-and-mitigations/

[144] OWASP GenAI Security Project. “OWASP Top 10 for Large Language Model Applications (Version 2025).” *OWASP*. November 2024. [Online]. Available: https://owasp.org/www-project-top-10-for-large-language-model-applications/

[145] MITRE Corporation. “MITRE ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems.” *MITRE*. December 2024. [Online]. Available: https://atlas.mitre.org

[146] M. Howard and S. Lipner. *The Security Development Lifecycle*. *Microsoft Press*. Volume 8. 2006.

[147] C. Lai and J. Spring. “Software Must Be Secure by Design, and Artificial Intelligence Is No Exception.” *CISA Blog*. 2023. [Online]. Available: https://www.cisa.gov/news-events/news/software-must-be-secure-design-and-artificial-intelligence-no-exception

[148] H. Booth, M. Souppaya, A. Vassilev, M. Ogata, M. Stanley, and K. Scarfone. “Secure Software Development Practices for Generative AI and Dual-Use Foundation Models: An SSDF Community Profile.” *NIST Special Publication* 800-218A. July 2024. [Online]. Available: https://doi.org/10.6028/NIST.SP.800-218A

[149] S. Longpre, K. Klyman, R. E. Appel, S. Kapoor, R. Bommasani, M. Sahar, S. McGregor, A. Ghosh, B. Blili-Hamelin, N. Butters, A. Nelson, D. A. Elazari, A. Sellars, C. J. Ellis, D. Sherrets, D. Song, H. Geiger, I. Cohen, L. McIlvenny, M. Srikumar, M. M. Jaycox, M. Anderljung, N. F. Johnson, N. Carlini, N. Miailhe, N. Marda, P. Henderson, R. S. Portnoff, R. Weiss, V. Westerhoff, Y. Jernite, R. Chowdhury, P. Liang, and A. Narayanan. “Position: In-House Evaluation Is Not Enough — Towards Robust Third-Party Evaluation and Flaw Disclosure for General-Purpose AI.” In *Forty-Second International Conference on Machine Learning Position Paper Track*. 2025.

[150] V. Mavroeidis and S. Bromander. “Cyber Threat Intelligence Model: An Evaluation of Taxonomies, Sharing Standards, and Ontologies within Cyber Threat Intelligence.” In *2017 European Intelligence and Security Informatics Conference*. Pages 91–98. 2017. [Online]. Available: https://doi.org/10.1109/EISIC.2017.20

[151] G. C. Bowker and S. L. Star. *Sorting Things Out: Classification and Its Consequences*. *MIT Press*. 2000.

[152] J. H. Saltzer and M. D. Schroeder. "The Protection of Information in Computer Systems." *Proceedings of the IEEE*. Volume 63, Issue 9, Pages 1278–1308. 1975.

[153] S. Abdelnabi and M. Fritz. "Adversarial Watermarking Transformer: Towards Tracing Text Provenance with Data Hiding." In *2021 IEEE Symposium on Security and Privacy*. Pages 121–140. 2021.

[154] R. Zhang, S. S. Hussain, P. Neekhara, and F. Koushanfar. "REMARK-LLM: A Robust and Efficient Watermarking Framework for Generative Large Language Models." In *33rd USENIX Security Symposium*. Pages 1813–1830. 2024.

[155] Q. Pang, S. Hu, W. Zheng, and V. Smith. "No Free Lunch in LLM Watermarking: Trade-Offs in Watermarking Design Choices." In *Advances in Neural Information Processing Systems*. Volume 37. Pages 138756–138788. 2024.

[156] K. Stouffer, T. Zimmerman, C. Tang, J. Lubell, J. Cichonski, and J. McCarthy. "Cybersecurity Framework Manufacturing Profile." *US Department of Commerce, National Institute of Standards and Technology*. 2017.

[157] E. Wallace, K. Xiao, R. Leike, L. Weng, J. Heidecke, and A. Beutel. "The Instruction Hierarchy: Training LLMs to Prioritize Privileged Instructions." *arXiv*. Volume abs/2404.13208. 2024.

[158] R. Laine, B. Chughtai, J. Betley, K. Hariharan, M. Balesni, J. Scheurer, M. Hobbhahn, A. Meinke, and O. Evans. "Me, Myself, and AI: The Situational Awareness Dataset (SAD) for LLMs." In *Advances in Neural Information Processing Systems*. Volume 37, Pages 64010–64118. 2024.

[159] The MITRE Corporation. "DevSecOps Best Practices Guide." *MITRE*. Technical Report PR_23-02103-1. June 2023.

[160] Y. Shavit, S. Agarwal, and M. Brundage. "Practices for Governing Agentic AI Systems." *OpenAI*. December 2023. [Online]. Available: https://openai.com/index/practices-for-governing-agentic-ai-systems/

[161] M. L. Tanke, M. Roy, N. Sabbineni, and M. Sunkara. "Best Practices for Building Robust Generative AI Applications with Amazon Bedrock Agents – Part 2." *Amazon Web Services*. October 2024. [Online]. Available: https://aws.amazon.com/blogs/machine-learning/best-practices-for-building-robust-generative-ai-applications-with-amazon-bedrock-agents-part-2/

[162] N. Emadamerho-Atori. "AI Guardrails in Agentic Systems Explained." *AltexSoft*. 2024. [Online]. Available: https://www.altexsoft.com/blog/ai-guardrails/

[163] OpenAI. "A Practical Guide to Building Agents." *OpenAI*. 2024. [Online]. Available: https://cdn.openai.com/business-guides-and-resources/a-practical-guide-to-building-agents.pdf

[164] Anthropic. "Frontier Model Security." *Anthropic*. 2024. [Online]. Available: https://www.anthropic.com/news/frontier-model-security

[165] OWASP Foundation. "OWASP Application Security Verification Standard (ASVS) Version 5.0.0." *OWASP*. May 2025. [Online]. Available: https://owasp.org/www-project-application-security-verification-standard/

[166] D. Hardt. “The OAuth 2.0 Authorization Framework.” *IETF RFC 6749*. October 2012. [Online]. Available: https://doi.org/10.17487/RFC6749

[167] K. Scarfone and P. Mell. “Intrusion Detection and Prevention Systems.” In *Handbook of Information and Communication Security*. Pages 177–192. Springer, 2010.

[168] P. Brangetto, E. Çalişkan, and H. Rõigas. “Cyber Red Teaming.” *NATO Cooperative Cyber Defence Centre of Excellence (CCDCOE)*. Volume 99. Page 100. 2015.

[169] N. Carlini and M. Nasr. “Remote Timing Attacks on Efficient Language Model Inference.” *arXiv*. Volume abs/2410.17175. 2024.

[170] J. McHugh, K. Šekrst, and J. Cefalu. “Prompt Injection 2.0: Hybrid AI Threats.” *arXiv*. Volume abs/2507.13169. 2025.

[171] S. Banerjee, P. Sahu, M. Luo, A. Vahldiek-Oberwagner, N. J. Yadwadkar, and M. Tiwari. “SoK: A Systems Perspective on Compound AI Threats and Countermeasures.” *arXiv*. Volume abs/2411.13459. 2024.

[172] Joint Task Force. “Risk Management Framework for Information Systems and Organizations.” *NIST Special Publication.* Number 800-37. 2018.

[173] AI Agent. “AI Agents: Get More Done Without Doing More.” [Online]. Available: https://aiagent.app/

[174] M. Palmer. “Secure Vibe Coding: The Tools and Fundamentals to Vibe Code Securely.” 2025. [Online]. Available: https://blog.replit.com/16-ways-to-vibe-code-securely

[175] Cursor. “Documentation: Ignore Files.” [Online]. Available: https://cursor.com/docs/context/ignore-files

[176] Salesforce. “Why Agentforce?” [Online]. Available: https://www.salesforce.com/agentforce/why/

[177] Anthropic. “Claude Code: Best Practices for Agentic Coding.” 2025. [Online]. Available: https://www.anthropic.com/engineering/claude-code-best-practices

[178] OpenAI. “ChatGPT Agent System Card.” *OpenAI*. July 2025. [Online]. Available: https://cdn.openai.com/pdf/6bcccca6-3b64-43cb-a66e-4647073142d7/chatgpt_agent_system_card_launch.pdf

[179] Anthropic. “Claude Opus 4 & Claude Sonnet 4 System Card.” *Anthropic*. May 2025. [Online]. Available: https://www.anthropic.com/claude-4-system-card

[180] Factory. “Building AI for Safe and Responsible Autonomy.” 2025. [Online]. Available: https://www.factory.ai/news/safe-autonomy-readiness-policy

[181] AICPA. “Guide: SOC 2 Reporting on an Examination of Controls at a Service Organization Relevant to Security, Availability, Processing Integrity, Confidentiality, or Privacy.” John Wiley & Sons. 2018.

[182] Scribe. “Security and Privacy at Scribe.” [Online]. Available: https://scribehow.com/security

[183] Simple AI. “Simple AI - Trust Center.” [Online]. Available: https://trust.usesimple.ai/

[184] 11x.ai. “11x Security Overview.” 2025. [Online]. Available: https://www.11x.ai/security

[185] Bardeen. “Security, Privacy & Compliance.” [Online]. Available: https://www.bardeen.ai/security

[186] OpenAI. "OpenAI o1 System Card." *OpenAI*. December 2024. [Online]. Available: https://openai.com/index/openai-o1-system-card/

[187] Amazon Web Services. "Security in Amazon Q Developer." AWS Documentation. 2025. [Online]. Available: https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/security.html

[188] Salesforce. "Best Practices for Secure Agentforce Implementation." *SalesForce Blog*. 2025. [Online]. Available: https://www.salesforce.com/blog/best-practices-for-secure-agentforce-implementation/

[189] A. Tamkin, M. McCain, K. Handa, E. Durmus, L. Lovitt, A. Rathi, S. Huang, A. Mountfield, J. Hong, S. Ritchie, M. Stern, B. Clarke, L. Goldberg, T. R. Sumers, J. Mueller, W. McEachen, W. Mitchell, S. Carter, J. Clark, J. Kaplan, and D. Ganguli. "Clio: Privacy-Preserving Insights into Real-World AI Use." *arXiv* Volume abs/2412.13678. 2024.

[190] A. Chatterji, T. Cunningham, D. J. Deming, Z. Hitzig, C. Ong, C. Y. Shan, and K. Wadman. "How People Use ChatGPT." *National Bureau of Economic Research*. Working Paper 34255. September 2025. [Online]. Available: https://doi.org/10.3386/w34255

[191] Z. Dong, Z. Zhou, C. Yang, J. Shao, and Y. Qiao. "Attacks, Defenses and Evaluations for LLM Conversation Safety: A Survey." In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. June 2024. Pages 6734–6747.

[192] B. C. Das, M. H. Amini, and Y. Wu. "Security and Privacy Challenges of Large Language Models: A Survey." *ACM Computing Surveys.* Volume 57. Number 6. Pages 1–39. 2025.

[193] Y. Yao, J. Duan, K. Xu, Y. Cai, Z. Sun, and Y. Zhang. "A Survey on Large Language Model (LLM) Security and Privacy: The Good, the Bad, and the Ugly." *High-Confidence Computing.* Volume 4. Number 2. Page 100211. 2024.

[194] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu. "A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions." *ACM Transactions on Information Systems.* Volume 43. Number 2. Pages 1–55. 2025.

[195] M. Costa, B. Köpf, A. Kolluri, A. Paverd, M. Russinovich, A. Salem, S. Tople, L. Wutschitz, and S. Zanella-Béguelin. "Securing AI Agents with Information-Flow Control." *arXiv* Volume abs/2505.23643. 2025.

[196] E. Debenedetti, I. Shumailov, T. Fan, J. Hayes, N. Carlini, D. Fabian, C. Kern, C. Shi, A. Terzis, and F. Tramèr. "Defeating Prompt Injections by Design." *arXiv*. Volume abs/2503.18813. 2025.

[197] L. Beurer-Kellner, B. Buesser, A.-M. Creţu, E. Debenedetti, D. Dobos, D. Fabian, M. Fischer, D. Froelicher, K. Grosse, D. Naeff, E. Ozoani, A. Paverd, F. Tramèr, and V. Volhejn. "Design Patterns for Securing LLM Agents Against Prompt Injections." *arXiv*. Volume abs/2506.08837. 2025.

[198] Joint Task Force. "Security and Privacy Controls for Information Systems and Organizations." *NIST Special Publication*. Number 800-53. 2017.

[199] Google DeepMind Security & Privacy Research Team. "Advancing Gemini's Security Safeguards." 2024. *Google DeepMind Blog*. [Online]. Available: https://deepmind.google/discover/blog/advancing-geminis-security-safeguards/

[200] S. Agashe, J. Han, S. Gan, J. Yang, A. Li, and X. E. Wang. “Agent S: An Open Agentic Framework That Uses Computers Like a Human.” *arXiv*. Volume abs/2410.08164. 2024.

[201] Amazon Web Services. “Amazon Q Developer.” [Online]. Available: https://aws.amazon.com/q/developer/

[202] Anthropic. “Introducing Computer Use, a New Claude 3.5 Sonnet, and Claude 3.5 Haiku.” 2025. [Online]. Available: https://www.anthropic.com/news/3-5-models-and-computer-use

[203] Cognition Labs. “Cognition Trust Center.” [Online]. Available: https://trust.cognition.ai/

[204] Anthropic. “Activating AI Safety Level 3 Protections.” 2025. [Online]. Available: https://www.anthropic.com/activating-asl3-report

[205] OpenAI. “Computer-Using Agent.” 2025. [Online]. Available: https://openai.com/index/computer-using-agent/

[206] OpenAI. “Operator System Card.” 2025. [Online]. Available: https://openai.com/index/operator-system-card/

[207] Amazon Web Services. “Amazon Q Business: Security and Governance.” [Online]. Available: https://aws.amazon.com/q/business/features/#topic-2

[208] Amazon Web Services. “Admin Controls and Guardrails in Amazon Q Business.” [Online]. Available: https://docs.aws.amazon.com/amazonq/latest/qbusiness-ug/guardrails.html

[209] Factory. “Code Droid: A Technical Report.” 2024. [Online]. Available: https://www.factory.ai/news/code-droid-technical-report

[210] H2O.ai. “Model Validation.” [Online]. Available: https://h2o.ai/platform/enterprise-h2ogpte/model-validation/

[211] H2O.ai. “H2O GPTe Enterprise Documentation.” Online documentation. 2025. [Online]. Available: https://docs.h2o.ai/enterprise-h2ogpte/tutorials/tutorial-7

[212] GitHub. “About Copilot Coding Agent.” [Online]. Available: https://docs.github.com/en/copilot/concepts/about-copilot-coding-agent

[213] Amazon Web Services. “Developing Features with Amazon Q Developer.” [Online]. Available: https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/software-dev.html

[214] A. Fourney, G. Bansal, H. Mozannar, V. Dibia, and S. Amershi. “Magentic-One: A Generalist Multi-Agent System for Solving Complex Tasks.” *Microsoft Research*. 2024. [Online]. Available: https://www.microsoft.com/en-us/research/articles/magentic-one-a-generalist-multi-agent-system-for-solving-complex-tasks/

[215] Sakana AI. “The Darwin Gödel Machine: AI That Improves Itself by Rewriting Its Own Code.” 2024. [Online]. Available: https://sakana.ai/dgm/

[216] Technion Kishony Lab. “Backward-Traceable AI-Driven Research.” 2024. [Online]. Available: https://github.com/Technion-Kishony-lab/data-to-paper?tab=readme-ov-file

[217] H2O.ai. “H2OGPTe Agentic AI Converges Generative AI and Predictive with Purpose-Built SLMs.” [Online]. Available: https://h2o.ai/platform/enterprise-h2ogpte/

[218] Replit. “Replit Auth.” [Online]. Available: https://docs.replit.com/replit-workspace/replit-auth

[219] Pythagora AI. “Your Apps, Your Security.” [Online]. Available: https://www.pythagora.ai/security

[220] Grit. “Data Security Overview.” [Online]. Available: https://docs.grit.io/security

[221] Codebuff. “Codebuff Best Practices.” [Online]. Available: https://www.codebuff.com/docs/tips#codebuff-best-practices

[222] Kodu AI. “AI Coding Agent Right in Your IDE.” [Online]. Available: https://www.kodu.ai/extension

[223] Salesforce. “Agentforce MCP Support.” [Online]. Available: https://www.salesforce.com/agentforce/mcp-support/

[224] Cursor. “Security.” 2025. [Online]. Available: https://cursor.com/security

[225] Simular. “Introducing Simular.” 2025. [Online]. Available: https://www.simular.ai/articles/introducing-simular

[226] Sakana AI. “AI Scientist GitHub.” 2024. [Online]. Available: https://github.com/SakanaAI/AI-Scientist?tab=readme-ov-file

[227] AllHands. “AllHands Documentation: Safety and Security.” [Online]. Available: https://docs.all-hands.dev/usage/faqs#safety-and-security

[228] AllHands. “Runtime Architecture.” [Online]. Available: https://docs.all-hands.dev/usage/architecture/runtime

[229] OpenAI. “Deep Research System Card.” 2025. [Online]. Available: https://openai.com/index/deep-research-system-card/

[230] Weco AI. “Weco AI FAQ.” [Online]. Available: https://docs.weco.ai/faq#privacy--security

[231] Anthropic. “Anthropic’s Transparency Hub: Model Report.” 2025. [Online]. Available: https://www.anthropic.com/transparency/model-report

[232] Replit. “Security Scanner.” [Online]. Available: https://docs.replit.com/replit-workspace/workspace-features/security-scanner#security-scanner

[233] Trase Systems. “AI, Uncomplicated.” [Online]. Available: https://www.trasesystems.com/

[234] Salesforce. “Accelerating Time to Agentic Value.” 2025. [Online]. Available: https://www.salesforce.com/en-us/wp-content/uploads/sites/4/documents/research/valoir-report-accelerating-agentic-ai-time-to-value.pdf

[235] SuperAGI. “SuperAGI SuperCoder.” 2023. [Online]. Available: https://superagi.com/supercoder/

Contact Us

Software Engineering Institute
4500 Fifth Avenue, Pittsburgh, PA 15213-2612
Phone: 412.268.5800 | 888.201.4479
Web: www.sei.cmu.edu
Email: info@sei.cmu.edu