

The Dual-Use Nature of Large Language Models in Cybersecurity: A Bibliometric Analysis and Critical Review

Muhammad Sharjeel Aziz

Niceb399@gmail.com

RIGA Technical University, Latvia(Liepaja Campus)

Muhammad Ramzan Ali

Ramzan.ali@ajou.uz

AJOU University in Tashkent, Tashkent, Uzbekistan <https://orcid.org/0000-0002-6510-4877>

Adnan Manzoor

Central Asian University, Tashkent, Uzbekistan

Muhammad Younis

RIGA Technical University, Latvia(Liepaja Campus)

Zulfiya Khabirova

AJOU University in Tashkent, Tashkent, Uzbekistan

Systematic Review

Keywords: Large language models, cybersecurity, dual-use technology, prompt injection, jailbreak, agentic AI, bibliometric analysis, VOSviewer, co-word analysis, co-citation analysis, security evaluation

Posted Date: October 29th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11434787/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

The Dual-Use Nature of Large Language Models in Cybersecurity: A Bibliometric Analysis and Critical Review

1st Muhammad Ramzan Ali, 2nd Adnan Manzoor

3rd Muhammad Sharjeel Aziz, 4th Muhammad Younis, 5th Zulfiya Khabirova

¹Assistant Professor, AJOU University in Tashkent, Tashkent, Uzbekistan (Ramzan.ali@ajou.uz)

²Associate Professor, Central Asian University, Tashkent, Uzbekistan (a.manzoor@centralasian.uz)

³rd Research Scholar at RIGA Technical University, Latvia(Liepaja Campus)(Sharjeel9200@gmail.com)

⁴th Research Scholar at RIGA Technical University, Latvia(Liepaja Campus)(Younischauhan44@gmail.com)

⁵th Associate Professor, AJOU University in Tashkent, Tashkent, Uzbekistan (zulya.uz@gmail.com)

Corresponding author: Muhammad Ramzan Ali (Ramzan.ali@ajou.uz)

<https://orcid.org/0000-0002-6510-4877>

Abstract: Large language models (LLMs) now occupy three positions in cybersecurity at once: they are systems that must be secured, capabilities that lower the cost of offence, and instruments that augment defence. Existing reviews typically examine one of these roles in isolation, which obscures how the roles interact and leaves the field without an evidence base for judging whether the interactions are being studied. This article addresses that gap by combining a VOSviewer-based bibliometric analysis with a critical literature review. We retrieved 13,756 records published between January 2023 and 19 October 2026 from OpenAlex and, after document-type, language, abstract, de-duplication and validated relevance screening (precision 84.0%, 95% CI 78.3–88.4%), analysed 7,369 documents through performance analysis, keyword co-occurrence and co-citation mapping in VOSviewer and a dictionary-based classification of each document's dual-use role. The co-occurrence map resolves into four thematic clusters: attacks on and alignment of LLMs; LLM-enabled detection and security operations; code and vulnerability analysis, including penetration testing; and socio-technical threats and governance. Research on LLMs as targets (42.8% of documents) and as defensive tools (35.3%) has grown largely in parallel: only 7.9% of documents address more than one role, and only 11.1% of defensive studies engage with attacks on the defending model itself, falling to 2.5% for vulnerability detection. The critical review explains why this matters: reported success rates of offensive LLM agents range from 0% to 90% depending on experimental design, few defences are evaluated against adaptive adversaries, and the security–utility trade-off is rarely quantified. We formalise these findings in a coupled dual-use framework with three testable propositions and derive eight research directions, prioritising adaptive evaluation, multilingual guardrail robustness, longitudinal measurement and the security of defensive LLM deployments.

Keywords: Large language models, cybersecurity, dual-use technology, prompt injection, jailbreak, agentic AI, bibliometric analysis, VOSviewer, co-word analysis, co-citation analysis, security evaluation

1. Introduction

The cybersecurity landscape has shifted at a rapid pace compared to other similar technology since the public release of instruction-tuned large language models (LLMs) in late 2022. Three lines of research have been documented in the months that followed, and the literature that followed has developed on all three. Three different effects were documented within months and the literature since has developed on all three at the same time. In the first 10 months of 2026, our bibliometric corpus grew from 267 documents in 2023 to 3,303 documents, with the documents explicitly related to security challenges.

The first one is that the LLM is now an attack surface. The inability to distinguish between instructions and data in a model that follows an instruction given in natural language is not necessarily due to an

implementation error, it is a property of the way these systems work (Greshake et al., 2023; Das et al., 2025). The attackers leveraged this property first through jailbreak prompts that are not aligned with safety, rather than aligned with safety (Wei et al., 2023; Shen et al., 2024), and then through prompt injection (Perez and Ribeiro, 2022; Greshake et al., 2023; Liu et al., 2023b), which involves injecting malicious instructions into content the model reads, rather than typing. The second phenomenon is the decrease in the cost of offence. Previously, it was possible to only partially automate such tasks, but with the help of artificial intelligence, these tasks can now be fully automated or, in part, completed with the aid of artificial intelligence (Fang et al., 2024a; Heiding et al., 2024; Schmitt and Flechais, 2024). The question here is not so much whether LLM's help attackers, as how much and how often. The third effect is that LLM's are used to enhance defence. Security operations centres (SOCs) are inundated with alerts and unstructured threat reporting, making models suited to reading, summarising, and structuring text at scale suitable for threat intelligence extraction, log analysis, detection-rule authoring, and incident-reporting (Chen et al., 2024; Xu et al., 2025).

These effects are not independent effects. An automated jailbreak is an attack tool, an attack on a model, and a target in a SOC deployed in a defensive LLM; you can turn the defensive LLM into the attack tool in a matter of months by utilizing a research paper you've read; an attack on a model is a target. This interdependence is captured by the long-standing notion of dual-use technology (Brundage et al., 2018) and by the offence defence balance literature, which asks how a new capability shifts the relative cost of attack and defence (Garfinkel and Dafoe, 2019). Yet the scholarly literature has largely partitioned along the three lines. Surveys of attacks on LLMs rarely treat LLMs as attack tools (Yi et al., 2024; Das et al., 2025), and surveys of LLMs for cyber defence rarely engage with the vulnerabilities of the defending model (Chen et al., 2024; Xu et al., 2025). More importantly, no prior review has measured the extent of this partition. Claims that the roles are "siloeed" have so far been asserted on the basis of the authors' reading rather than demonstrated from the structure of the literature.

This article responds with a two-stage design that combines a VOSviewer-based bibliometric analysis (van Eck and Waltman, 2010; Donthu et al., 2021) with a critical literature review. The bibliometric stage maps the conceptual structure (keyword co-occurrence), intellectual structure (co-citation) and role composition of 7,369 documents; the critical stage interrogates the evidence within each role, focusing on whether reported results are comparable and robust. The two stages are linked by design: the bibliometric clusters determine the organisation of the critical review, and the critical review supplies the interpretive judgement that bibliometric counts cannot.

The study is guided by four research questions, detailed in Section 4.2:

- **RQ1.** How has research at the intersection of LLMs and cybersecurity developed in volume, venues, geography and influence between 2023 and 2026?
- **RQ2.** What are the principal thematic clusters and intellectual foundations of the field, and how are they evolving?
- **RQ3.** To what extent does the literature study the target, weapon and shield roles of LLMs jointly rather than in isolation?
- **RQ4.** How robust is the empirical evidence on threats and defences within each role, and which research directions follow from the combined bibliometric and critical evidence?

The article makes five contributions:

- A coupled dual-use framework (Section 2.3, Figure 1) that conceptualises LLMs as target, weapon and shield and specifies three coupling propositions that can be tested empirically.
- A transparent, reproducible VOSviewer mapping of 7,369 documents, including validated screening, a benchmark-based recall test, sensitivity analysis of the clustering solution, and released VOSviewer map and network files.

- The first quantitative evidence of role partitioning in the field: only 7.9% of documents span more than one role, and only 11.1% of defensive studies engage with attacks on the defending model.
- A three-layer taxonomy of attacks on LLM-based systems (model, system and agentic layers) and a critical synthesis showing that reported offensive and defensive effectiveness varies by an order of magnitude with experimental design.
- A research agenda of eight directions, each linked to bibliometric and critical evidence and assessed for feasibility by research groups without frontier-scale compute.

The remainder of the article is organised as follows. Section 2 presents the background and theoretical framework. Section 3 reviews related surveys and states the research gap. Section 4 describes the methodology. Section 5 reports the bibliometric results. Section 6 presents the critical literature analysis organised by role. Section 7 discusses the findings, implications and limitations, and Section 8 concludes.

2. Background and theoretical framework

2.1 Large language models and agentic systems

LLMs are transformer-based models trained with a wealth of text corpora and then fine-tuned by a combination of instruction tuning and preference optimisation to accept natural language instructions (Brown et al., 2020). There are two architectural changes that are important for security. Retrieval-augmented generation (RAG) is a method that adds external documents to the output context window of models to enhance the factuality of the output, but also introduces untrusted content into the context window of the model. Agentic systems take this further by providing the model with persistent memory, invoking tools, planning multi-step actions, and communicating with other agents (He et al., 2025). Each extension adds capability and at the same time attack surface. The NIST AI Risk Management Framework defines AI systems as socio-technical, with risks stemming from the technology and from how people interact with it (NIST, 2023), and the NIST adversarial machine learning taxonomy includes the classical categories of attack, namely evasion, poisoning, privacy and abuse, which is extended to generative AI by Vassilev et al. (2024).

2.2 Dual-use technology and the offence–defence balance

Dual-use technologies are technologies that can be used for both good and bad. AI increases the threat from reducing the cost of attacks, presents new threats, and alters the nature of threat by making them more targeted and difficult to attribute, according to Brundage et al. (2018). The other lens that can be brought to bear is drawn from the literature on offence and defence, which suggests that a technology changes the balance of attack and defence by altering the relative cost, and that the direction of the shift in the balance may be dependent on the scale of what is invested (Garfinkel and Dafoe, 2019). A lack of properly designed incentives, not just technical constraints, is often the deciding factor in the success or failure of security (Anderson and Moore, 2006). These theories collectively predict that (i) attackers and defenders will learn to use an LLM at the same time, (ii) adversarial attacks will emerge against the LLM as soon as it is deployed in valuable contexts, and (iii) the offensive or defensive nature of research will depend on the user of the LLM, rather than the technique. The adversarial machine learning tradition also teaches that defenses against non-adaptive attackers systematically overestimate robustness (Athalye et al., 2018; Carlini et al., 2019); we explore the implications of this lesson for security of LLM in Section 6.

2.3 The coupled dual-use framework

Building on this theory, we organise the field around three relationships between LLMs and cybersecurity (Figure 1): the LLM as target (the model, the application around it and the agent built on it are attacked), the LLM as weapon (the model is used to attack other systems or people), and the LLM as shield (the model is used to protect other systems). The framework's distinctive element is not the triad itself, which echoes

earlier "good, bad and ugly" framings (Gupta et al., 2023; Yao et al., 2024), but three coupling propositions that connect the roles:

- C1 (shield → target). Every defensive LLM deployment creates a new target. A threat-intelligence copilot that ingests open-source reports is structurally a retrieval pipeline consuming attacker-controllable text, and is therefore exposed to indirect prompt injection (Greshake et al., 2023; Yi et al., 2025). If the literature were integrated, defensive studies would routinely evaluate the defending model's own robustness.
- C2 (weapon → shield). Offensive capability is re-purposed as defence. Autonomous exploitation agents and vulnerability-discovery systems perform closely related reasoning over the same artefacts, so the same research serves red teaming and vulnerability management (Happe and Cito, 2023; Wang et al., 2025).
- C3 (target → weapon). Research on attacking LLMs produces attack tooling. Automated jailbreak generators and universal injection payloads are, once published, instruments for misuse as much as diagnostic tools (Zou et al., 2023; Deng et al., 2024; Chao et al., 2025).

The propositions yield an empirical expectation that Section 5 tests: if research were organised around the coupled system, a substantial share of documents would address more than one role, and defensive studies would routinely examine attacks on the defending model. Low overlap would indicate that the coupling exists in the technology but not in the research.

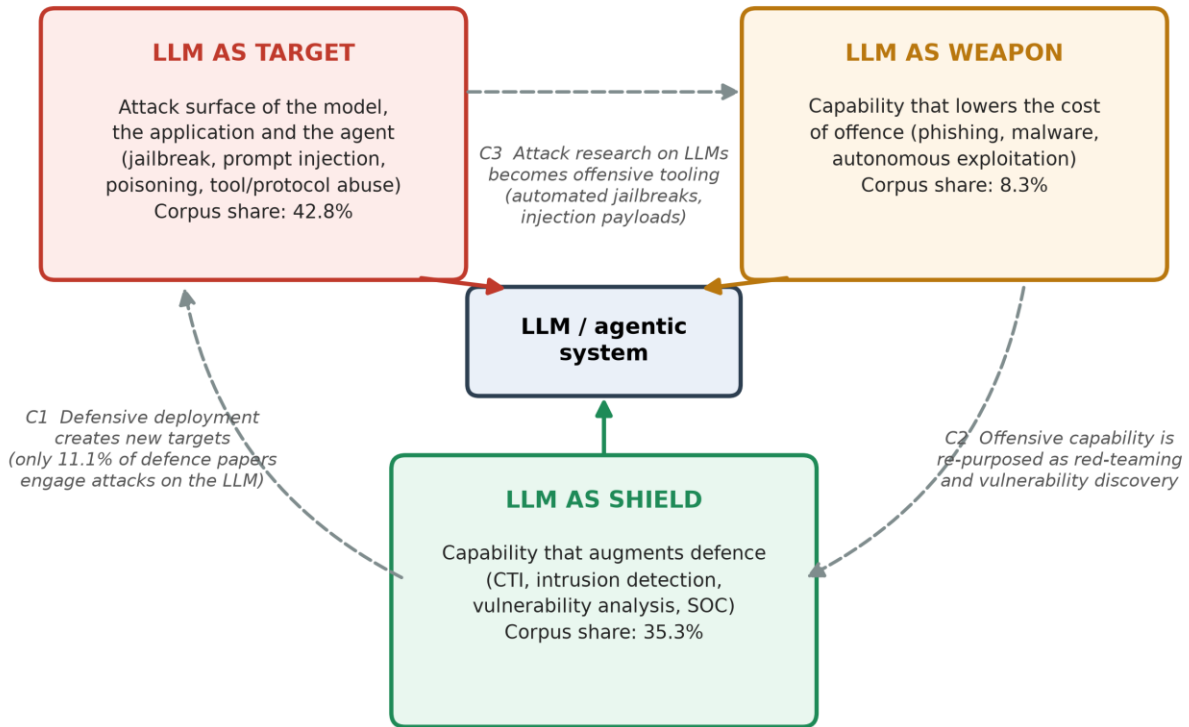


Figure 1. Coupled dual-use framework. Solid arrows denote the three relationships between LLM-based systems and cybersecurity; dashed arrows denote the coupling propositions C1–C3. Corpus shares are the proportions of the 7,369 analysed documents classified into each role (Section 4.9); documents may belong to more than one role.

2.4 A three-layer view of the attack surface

To analyse the target role consistently, we adopt a three-layer model of where a vulnerability resides (Figure 2). Model-layer threats target weights or alignment independently of any application; system-layer threats target the application assembled around the model, especially the concatenation of trusted instructions and untrusted data in a single context window; and agentic-layer threats exploit memory, tools, planning and

inter-agent communication. Persistence and potential blast radius increase with height: model- and system-layer attacks are often confined to a single turn, whereas agentic-layer attacks can persist across sessions. The layering is adapted from the layered attack-surface model of Chu (2026) and the autonomy-centred analysis of Su et al. (2025) and is used throughout Section 6.

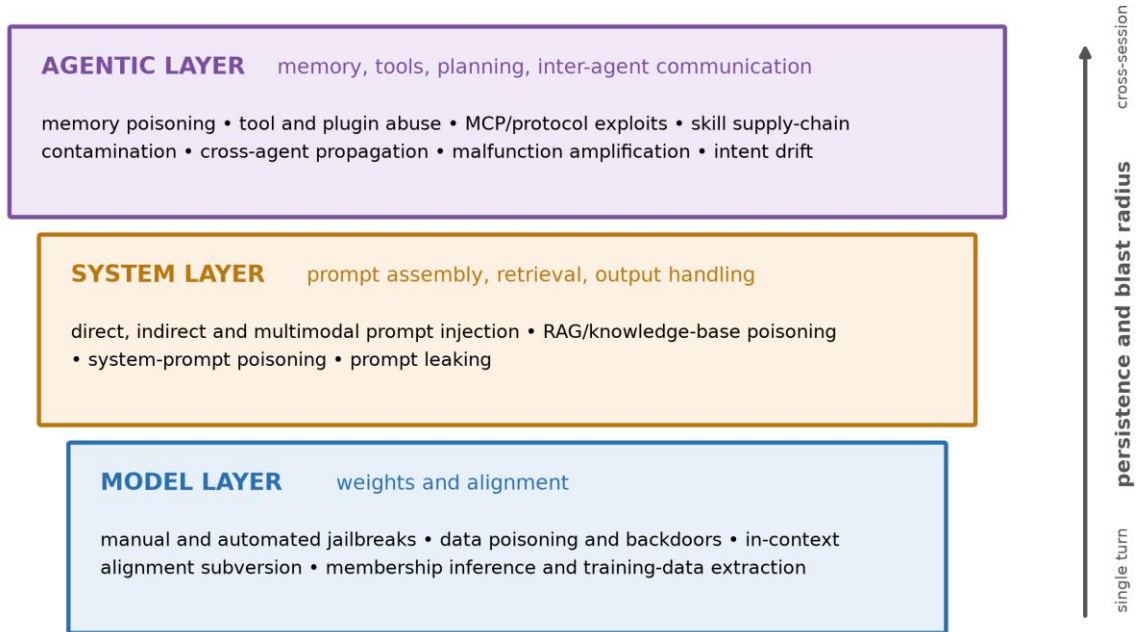


Figure 2. Three-layer attack surface of LLM-based systems. The layer structure is adapted from Chu (2026) and Su et al. (2025); persistence and potential blast radius increase from the model layer to the agentic layer.

3. Related work and research gap

Table 1 compares this review with the principal existing surveys. Three patterns emerge. First, coverage of each individual role is now good: attacks on LLMs and their defences have been surveyed extensively (Yi et al., 2024; Das et al., 2025; Correia et al., 2026; Damacena Duarte et al., 2026), as have LLMs for cyber defence (Chen et al., 2024; Xu et al., 2025; Zhang et al., 2025) and agent security (He et al., 2025; Chhabra et al., 2026; Ferrag et al., 2026). Second, the few reviews that span all three roles do so descriptively, without a consistent taxonomy across roles and without testing whether the roles are studied together (Gupta et al., 2023; Yao et al., 2024; Ferrag et al., 2025). Third, almost all prior reviews are narrative or systematic syntheses of a curated set of papers; to our knowledge, none combines a large-scale science-mapping analysis with critical appraisal of the empirical evidence. Systematic reviews in the area typically include between several dozen and a few hundred studies (Xu et al., 2025; Zhang et al., 2025), which is appropriate for depth but cannot reveal the macro-structure of a field that now produces several thousand documents per year.

Table 1. Comparison with existing surveys (● covered in depth, ● partially covered, ○ not covered).

Survey	Venue	Target	Weapon	Shield	Agentic layer	Evaluation critique	Science mapping
From ChatGPT to ThreatGPT (Gupta et al., 2023)	IEEE Access	●	●	●	○	○	○
LLM security and privacy: the good, the bad and the ugly (Yao et al., 2024)	High-Confidence Computing	●	●	●	○	○	○
Security and privacy challenges of LLMs (Das et al., 2025)	ACM CSUR	●	○	○	○	●	○
LLMs for cyber security: an SLR (Xu et al., 2025)	ACM TOSEM	○	●	●	●	●	○

LLMs for cyber threat detection (Chen et al., 2024)	Computers & Security	○	○	●	○	○	○
When LLMs meet cybersecurity (Zhang et al., 2025)	Cybersecurity	●	●	●	○	●	○
Jailbreak attacks and defenses (Yi et al., 2024)	arXiv	●	○	○	○	●	○
Security and privacy of LLM agents (He et al., 2025)	ACM CSUR	●	○	○	●	○	○
Prompt injections to protocol exploits (Ferrag et al., 2026)	ICT Express	●	●	○	●	○	○
Generative AI in cybersecurity (Ferrag et al., 2025)	IoT and Cyber-Physical Systems	●	●	●	○	○	○
Agentic AI security (Chhabra et al., 2026)	IEEE Access	●	●	○	●	●	○
Systematic review of prompt injection (Damacena Duarte et al., 2026)	IEEE Access	●	○	○	●	●	○
SLR on LLM defences (Correia et al., 2026)	arXiv	●	○	○	●	●	○
This review	—	●	●	●	●	●	●

The resulting gap has three components. The first is structural: the field lacks an evidence-based account of whether its three roles are studied together, which is a precondition for judging claims about dual use. The second is evaluative: individual surveys note that results are hard to compare, but none systematically relates the variance in reported results to experimental design across roles. The third is methodological: bibliometric mapping and critical appraisal have developed as separate traditions (Zupic and Čater, 2015; Donthu et al., 2021), and their integration remains rare in security research. This review addresses all three by mapping the field at scale, measuring role coupling directly, and using the map to structure a critical appraisal of the evidence.

4. Methodology

4.1 Research design

We adopted a sequential two-stage design (Figure 3). Stage 1 is a science-mapping study following the bibliometric workflow of Zupic and Čater (2015) and the guidelines of Donthu et al. (2021): research design, data compilation, analysis, visualisation and interpretation. Record selection is reported in a PRISMA 2020-style flow diagram (Page et al., 2021) to make each exclusion transparent, although the bibliometric stage does not claim the exhaustive inclusion of a systematic review. Stage 2 is a critical literature review of a curated evidence set, organised by the clusters and roles identified in Stage 1. The evidence set was extended by backward and forward snowballing from the most influential and most locally cited documents. Integration occurs at three points: the clusters define the structure of Section 6; the co-citation analysis identifies foundational works that the critical review must address; and the role-coupling measures provide the quantitative test of the framework in Section 2.3.

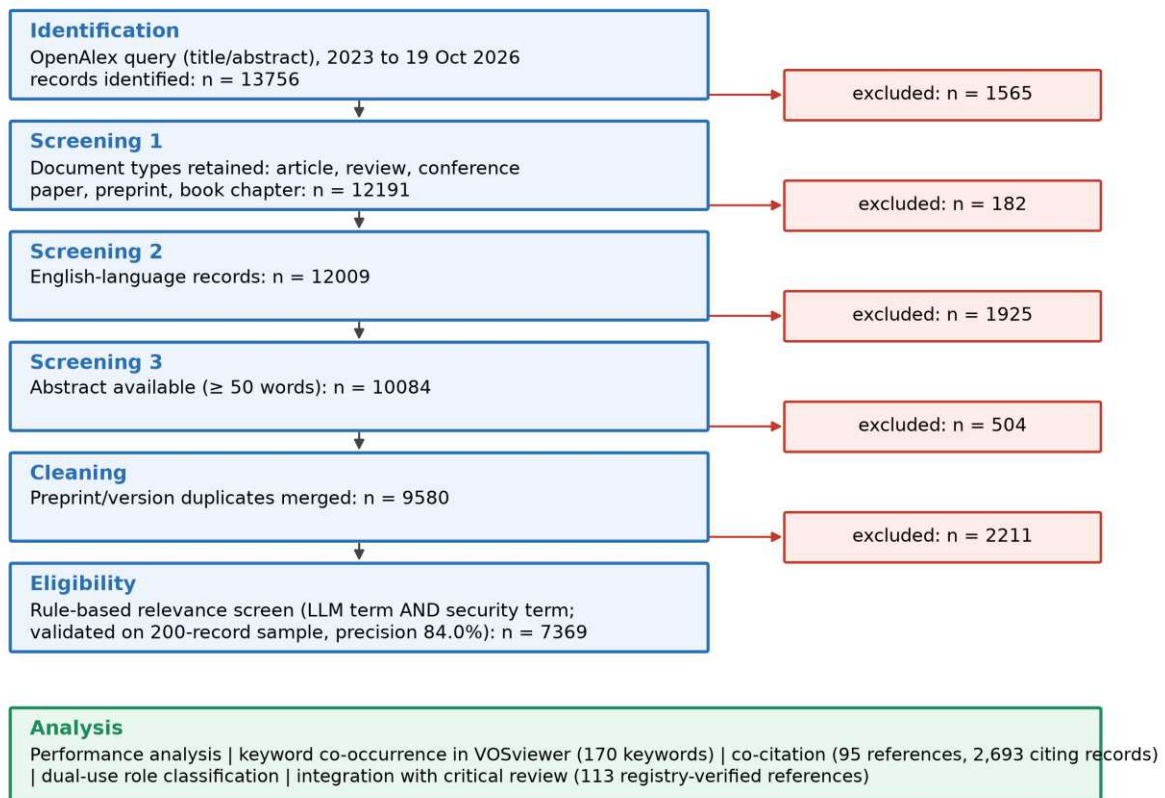


Figure 3. Research design and record selection. Counts are produced by the released scripts from the OpenAlex export of 19 October 2026.

4.2 Research questions

Table 2 maps each research question to its analytic technique and to the section that answers it.

Table 2. Research questions, techniques and corresponding sections.

RQ	Question	Technique	Section
RQ1	How has the field developed in volume, venues, geography and influence?	Performance analysis (publications, citations, sources, countries, influential documents)	5.1
RQ2	What are the field's thematic clusters and intellectual foundations, and how are they evolving?	Keyword co-occurrence (co-word) mapping, overlay analysis, co-citation analysis	5.2 to 5.4
RQ3	To what extent are the target, weapon and shield roles studied jointly?	Dictionary-based role classification, overlap and cross-cluster linkage measures	5.5
RQ4	How robust is the evidence within each role, and which research directions follow?	Critical literature review structured by Stage 1 clusters	6 and 7

4.3 Database selection

We used OpenAlex (Priem et al., 2022) as the bibliographic source for three reasons. First, coverage: research on LLM security is unusually dependent on preprints, and a large share of influential work including several of the most-cited attacks and benchmarks appears on arXiv before, or instead of, formal publication. Subscription databases such as Scopus and Web of Science index preprints poorly, so restricting the analysis to them would systematically omit a substantial part of the field. Second, Reproducibility: OpenAlex is openly accessible, which means that any reader can repeat the exact query and export, and VOSviewer can directly read OpenAlex data. Third, metadata richness: The metadata OpenAlex records include abstracts, algorithmically-assigned keywords, and reference lists and institutional affiliations, that all enable co-word and co-citation analysis and geographic analysis. There are, however, some drawbacks.

Comparative studies have noted that while coverage of works in OpenAlex is comprehensive, OpenAlex's reference lists and some metadata fields are less complete and less consistent than in curated databases (Alperin et al., 2024; Culbert et al., 2025). We addressed these limitations in the cleaning and validation and rectification process outlined in Sections 4.6 and 4.7, and we explicitly address them in Section 7.8.

4.4 Search strategy

Two concept blocks with Boolean operators were used and searched on the title and abstracts (Table 3). The first block was about technology (LLMs, generative AI, named model families), and the second block was about the security context in general and the primary security tasks and defence. The search was limited to the publication year 2023-2026 which was conducted on 19 October 2026, meaning the 2026 data are partial. Records were pulled year by year using the OpenAlex API, to stay within the paging limits.

Table 3. Search strategy.

Element	Specification
Database	OpenAlex (works endpoint), accessed through the public API
Field	Title and abstract (title and abstract.search)
Concept block A (technology)	"large language model" OR "large language models" OR LLM OR LLMs OR ChatGPT OR GPT-4 OR "generative AI" OR "foundation model"
Concept block B (security)	cybersecurity OR "cyber security" OR "prompt injection" OR jailbreak OR jailbreaking OR phishing OR malware OR "penetration testing" OR "intrusion detection" OR "threat intelligence" OR "vulnerability detection" OR "adversarial attack"
Combination	A AND B
Period	Publication years 2023–2026 (search executed 19 October 2026)
Records identified	13,756 unique works

4.5 Inclusion and exclusion criteria

The inclusion criteria were: (IC1) the document is a journal article, review, conference paper, preprint or book chapter; (IC2) the title or abstract mentions an LLM or closely related generative model; (IC3) the title or abstract mentions a security threat, attack, defence or security task; (IC4) at least one of these two concepts is in the title, not just incidental. Documents omitted from the study due to one or more of the following reasons were classified as exclusion criteria: (1) datasets and software records, editorials, errata, peer-review reports, paratexts, dissertations, reports and other non-research document types, (2) documents not written in English, (3) documents lacking an abstract of 50 words or more (required for relevance screening and text-based analysis), (4) duplicate versions of the same work.

4.6 Data cleaning

Data cleaning proceeded in five steps. (1) Abstracts were reconstructed from the OpenAlex inverted index. (2) Records were de-duplicated by normalised title; where a preprint and a published version coexisted, the published version was retained, the higher citation count was kept and reference lists were merged, reducing 10,084 records to 9,580. (3) Keyword variants were unified through a thesaurus file (Appendix B), for example merging "jailbreak attacks", "jailbreaking" and "LLM jailbreaking" into "jailbreak". (4) Generic terms that are implied by the search query and therefore connect to almost every node "large language models", "LLMs", "generative AI", "artificial intelligence", "machine learning", "deep learning" and "natural language processing" were removed from the co-occurrence analysis, following standard practice in co-word mapping (Callon et al., 1983; van Eck and Waltman, 2010). (5) Among the 100 most co-cited references, five OpenAlex records with demonstrably corrupted metadata (titles unrelated to their bibliographic identifiers) were excluded, and six records whose arXiv identifiers were correct but whose titles were corrupted were relabelled from their identifiers; for example, arXiv:2307.15043 was relabelled as the study by Zou et al. (2023). These corrections are listed in the supplementary material.

4.7 Screening validation

Relevance was screened with a transparent rule: a record was retained if its title or abstract contained a technology term and a security term, with at least one of them in the title (Appendix A). To estimate the precision of this rule, a random sample of 200 retained records (fixed seed) was screened manually against the inclusion criteria; 168 were judged relevant, giving a precision of 84.0% (Wilson 95% CI 78.3–88.4%). Most false positives were documents that mention LLMs and security only peripherally, such as generic applications of ChatGPT in other domains. To estimate recall, we used the 131 references of the critical evidence base as a benchmark set compiled independently of the query. The search retrieved 92 of the 131 (70.2%); after excluding 15 benchmark items outside the query's scope by design (one pre-2023 work, one general adversarial-machine-learning survey and 13 evaluation-methodology and governance studies), recall was 92 of 116 (79.3%). Missed in-scope items were concentrated in agent-security and secure-code-generation studies whose abstracts use terms such as "agent", "attack" or "secure code" without the security terms of block B. We therefore interpret the agentic and secure-code themes in the bibliometric maps as under-represented, and the critical review compensates by including these studies through snowballing.

4.8 Bibliometric techniques and parameters

Three complementary techniques were applied (Table 4). Performance analysis described annual output, document types, sources, countries and influential documents, using both global citations (OpenAlex citation counts) and local citations (citations received from other documents in the corpus). Keyword co-occurrence analysis mapped the conceptual structure of the field (Callon et al., 1983): two keywords co-occur when they are assigned to the same document, with binary counting per document. Co-citation analysis mapped the intellectual structure (Small, 1973): two references are co-cited when they appear together in the reference list of a corpus document, which reveals the knowledge base on which the field draws.

Both networks were analysed and visualised in VOSviewer version 1.6.21 (van Eck and Waltman, 2010). Co-occurrence and co-citation matrices were compiled from the cleaned OpenAlex records with Python scripts and imported into VOSviewer as map and network files; all normalisation, clustering, layout and visualisation steps were then performed by VOSviewer itself. Co-occurrence counts were normalised with the association strength measure, $s_{ij} = 2m \cdot c_{ij} / (k_i \cdot k_j)$, where c_{ij} is the number of co-occurrences, k_i is the total link strength of item i and m is the total link strength of the network; association strength is recommended over set-theoretic measures for bibliometric co-occurrence data (van Eck and Waltman, 2009). VOSviewer's clustering technique, which optimises a modularity-based quality function with the Leiden algorithm (Waltman et al., 2010; Traag et al., 2019), was run with resolution $\gamma = 1.0$, a minimum cluster size of eight (six for co-citation) and merging of small clusters. Item positions were computed with the VOS mapping technique (van Eck and Waltman, 2010; Waltman et al., 2010), with attraction 2 and repulsion 1. VOSviewer was run from the command line with these parameters so that the analysis is exactly repeatable; the input and output map and network files are released as supplementary material. As an independent check, the clustering was re-implemented in Python (python-igraph and leidenalg); its partition agreed closely with VOSviewer's (adjusted Rand index, ARI, 0.92).

Table 4. Bibliometric analysis parameters.

Analysis	Unit of analysis	Counting	Threshold	Normalisation	Clustering	Items / links / clusters
Keyword co-occurrence	OpenAlex keywords after thesaurus	Binary per document	≥ 40 occurrences	Association strength	VOSviewer (Leiden), $\gamma = 1.0$, min. size 8	170 / 6,803 / 4
Overlay	Same as above	—	—	—	—	Colour = mean publication date of documents containing the keyword

Co-citation	Cited references (2,693 citing records with reference lists)	Binary per citing document	100 most co-cited (≥ 29 citing documents); 5 corrupted records removed	Association strength	VOSviewer (Leiden), $\gamma = 1.0$, min. size 6	95 / 2,760 / 3
Role classification	Title and abstract	Dictionary match (Appendix A)	—	—	—	Target, weapon, shield, combinations

Clustering solutions in co-word analysis depend on parameter choices, so we tested robustness by re-running VOSviewer with alternative settings. Varying the resolution produced three clusters at $\gamma = 0.8$ (ARI relative to the reference solution 0.65), six at $\gamma = 1.2$ (ARI 0.68) and eight at $\gamma = 1.5$ (ARI 0.47). Varying the occurrence threshold produced five clusters with 232 items at a threshold of 30 (ARI on shared items 0.69) and four clusters with 102 items at a threshold of 60 (ARI 0.80). The four-cluster solution is therefore stable across reasonable thresholds; higher resolutions subdivide its clusters rather than reorganising them, which we take into account when interpreting cluster boundaries.

4.9 Dual-use role classification

To test the coupling propositions, every document was classified into zero or more of the three roles by matching its title and abstract against role dictionaries (Appendix A). The target dictionary captures attacks on and safeguards for LLMs (for example jailbreaks, prompt injection, poisoning, guardrails and red teaming); the weapon dictionary captures the use of LLMs for offence (for example generation of phishing or malware, offensive security and penetration testing, and autonomous exploitation); and the shield dictionary captures defensive applications (for example threat intelligence, intrusion and malware detection, vulnerability detection and repair, and incident response). Penetration testing was assigned to the weapon role because it operationalises offensive capability, although it serves defensive purposes; this choice makes the measured weapon–shield overlap a conservative estimate. Manual checks of random samples indicated precision of 24/25 for the target role, 23/25 for the shield role and 23/30 for the weapon role, after the weapon dictionary had been tightened to exclude generic uses of "misuse" and "attackers". Because abstract-level dictionaries cannot detect everything a full text contains, the coupling measures are interpreted as indicators of what studies foreground, not as exhaustive counts.

4.10 Critical review procedure

The critical evidence set was assembled from the 131 sources of the authors' earlier evidence base, the most influential and most locally cited documents identified in Stage 1, the foundational works identified by co-citation analysis and methodological sources on bibliometrics and security evaluation. To keep the synthesis focused, redundant surveys, superseded preprint versions and studies peripheral to the four clusters were then removed, leaving the 113 sources cited in this article. Every reference was verified against the Crossref or DataCite registry; bibliographic details were corrected where they differed from the registered record (Section 7.8). For each empirical study, we extracted the threat model, target system, evaluation setting, reported effect and qualifying conditions, and appraised the evidence against five criteria derived from the evaluation literature (Carlini et al., 2019; Arp et al., 2022; Jia et al., 2026): (a) whether the evaluation included adaptive attacks; (b) whether utility or false-positive costs were reported; (c) whether results were based on repeated trials; (d) whether targets were realistic (for example real CVEs rather than synthetic tasks); and (e) whether the study was peer reviewed. These criteria inform the qualifiers reported in Tables 12–14. Numerical results are reported as stated in the primary studies; we did not reproduce them.

4.11 Reproducibility

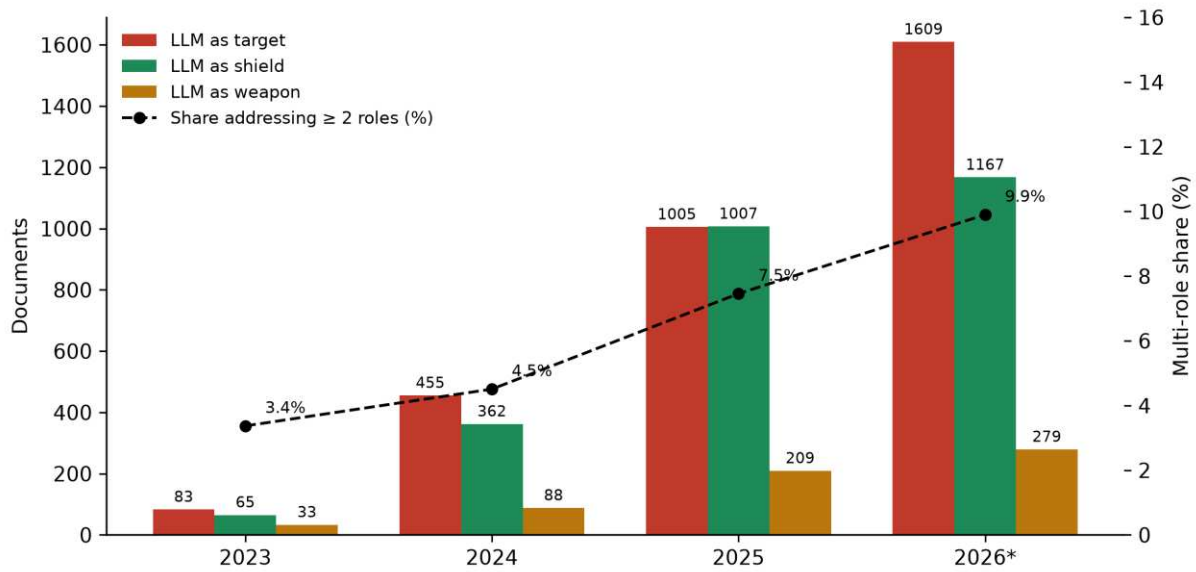
The query, harvesting, cleaning, screening sample with labels, thesaurus, role dictionaries, analysis scripts, VOSviewer map and network files, and the reference verification log are provided as supplementary

material. Re-running the harvest at a later date will return additional documents because OpenAlex is continuously updated; the counts reported here correspond to the export of 19 October 2026.

5. Bibliometric results

5.1 Performance analysis (RQ1)

The final corpus contains 7,369 documents by 21,696 distinct authors, which had received 36,520 citations in OpenAlex by 19 October 2026. Annual output grew from 267 documents in 2023 to 1,198 in 2024, 2,601 in 2025 and 3,303 in the first ten months of 2026 (Figure 4), roughly doubling or more in each full year. The document mix reflects a field that publishes rapidly and informally: journal articles account for 30.7% of the corpus (2,261), conference papers for 29.5% (2,172) and preprints for 35.9% (2,643), with reviews (88) and book chapters (205) making up the remainder. The preprint share is substantially higher than is typical of mature security sub-fields and is one of the structural explanations for the evidence problems discussed in Section 7.3.



*2026 covers 1 January to 19 October. Documents may carry more than one role; 1,631 records (22.1%) matched no role dictionary.

Figure 4. Annual output by dual-use role and share of documents addressing two or more roles. Roles are assigned by the dictionaries in Appendix A; a document can carry several roles.

Publication venues are dispersed (Table 5). arXiv is by far the largest single source (2,264 documents), followed by the Zenodo repository (426), whose records contribute few citations. Among peer-reviewed outlets, IEEE Access is the most productive and also the most cited journal source (119 documents; 3,047 citations), followed by major AI and natural-language-processing venues (AAAI, ACL, NeurIPS and EMNLP) and by software-engineering and security journals. The prominence of AI venues relative to security venues indicates that much of the research on attacks against LLMs is published in the machine-learning community, whereas defensive applications are spread across security and engineering journals.

Table 5. Most productive sources in the corpus.

Rank	Source	Documents	Citations
1	arXiv	2,264	3,222
2	Zenodo	426	19
3	IEEE Access	119	3,047
4	Proceedings of the AAAI Conference on Artificial Intelligence	76	496
5	Electronics	71	388
6	Annual Meeting of the Association for Computational Linguistics	54	375

7	Advances in Neural Information Processing Systems	52	750
8	Preprints.org	48	65
9	Conference on Empirical Methods in Natural Language Processing	48	263
10	Applied Sciences	34	200
11	Scientific Reports	29	111
12	ACM Transactions on Software Engineering and Methodology	28	574

Geographically, research is concentrated in a small number of countries (Table 6). Of the 5,292 documents with affiliation data, the United States contributes 1,648 and China 1,280, followed by India, the United Kingdom, Singapore, Australia, Canada and Germany. International collaboration is common (1,283 documents, 24.2% of those with affiliation data), and the strongest collaboration ties are China–United States (151 documents), China–Singapore (142) and China–Hong Kong (102). Citation impact is not proportional to output: Singapore, Australia and the United Arab Emirates achieve citation totals comparable to countries with considerably more documents. Central Asia is almost absent: Kazakhstan (10 documents), Uzbekistan (8) and Kyrgyzstan (2) together account for well under 1% of affiliated documents. This geographic concentration is directly relevant to the multilingual robustness gap discussed in Section 7.7, because the languages in which guardrails are tested follow the languages of the research communities that build them.

Table 6. Most productive countries (documents with at least one author affiliated with the country).

Rank	Country	Documents	Citations	Citations per document
1	United States	1,648	14,124	8.6
2	China	1,280	6,180	4.8
3	India	425	1,280	3.0
4	United Kingdom	317	2,789	8.8
5	Singapore	247	2,762	11.2
6	Australia	218	2,818	12.9
7	Canada	180	1,506	8.4
8	Germany	176	2,013	11.4
9	Hong Kong	165	1,025	6.2
10	Italy	155	803	5.2
11	South Korea	131	953	7.3
12	Saudi Arabia	121	768	6.3
13	Japan	120	511	4.3
14	United Arab Emirates	100	2,140	21.4
15	Pakistan	89	343	3.9

Table 7 lists the most influential documents. Because global citation counts favour broad surveys and older works, we report local citations alongside them. Three documents dominate both rankings: the survey of LLM security and privacy by Yao et al. (2024), the early mapping of generative AI’s offensive and defensive uses by Gupta et al. (2023), and the demonstration of indirect prompt injection against real-world LLM-integrated applications by Greshake et al. (2023), which is the most locally cited document in the corpus (170 citing documents). The remaining highly cited works are foundational attack studies Jailbroken (Wei et al., 2023), in-the-wild jailbreaks (Shen et al., 2024), the PAIR black-box jailbreak (Chao et al., 2025) and multi-step privacy jailbreaks (Li et al., 2023)—together with broad reviews (Ferrara, 2024; Das et al., 2025; Xu et al., 2025; Zhang et al., 2025) and a small number of defensive systems, notably a privacy-preserving BERT-based intrusion detector (Ferrag et al., 2024) and zero-shot vulnerability repair (Pearce et al., 2023). Influence is therefore concentrated in attack demonstrations and surveys; no evaluation-methodology study appears among the most influential documents, although such studies are among the most consequential for interpreting the rest of the literature (Section 6.5).

Table 7. Most influential documents in the corpus (global citations from OpenAlex; local citations from corpus documents; off-topic records removed).

Document	Role	Global citations	Local citations
Survey of LLM security and privacy (Yao et al., 2024)	Shield / target	1,066	146
From ChatGPT to ThreatGPT (Gupta et al., 2023)	Target / shield	792	131
Indirect prompt injection (Greshake et al., 2023)	Target	608	170
Security and privacy challenges of LLMs (Das et al., 2025)	Target	359	48
BERT-based lightweight threat detection (Ferrag et al., 2024)	Shield	316	86
Jailbroken (Wei et al., 2023)	Target	270	61
GenAI against humanity (Ferrara, 2024)	Weapon	260	—
In-the-wild jailbreak prompts (Shen et al., 2024)	Target	249	87
When LLMs meet cybersecurity (Zhang et al., 2025)	Review	234	86
Jailbreaking black-box LLMs in twenty queries (Chao et al., 2025)	Target	218	86
Zero-shot vulnerability repair (Pearce et al., 2023)	Shield	205	53
Multi-step jailbreaking privacy attacks (Li et al., 2023)	Target	203	46
LLMs for cyber security: an SLR (Xu et al., 2025)	Shield	—	68
Penetration testing with LLMs (Happe and Cito, 2023)	Weapon	—	61

5.2 Conceptual structure: keyword co-occurrence (RQ2)

The keyword co-occurrence network contains 170 keywords that occur in at least 40 documents, connected by 6,803 links. VOSviewer's clustering yields four clusters (Figure 5; Table 8), which together define the thematic structure that organises the critical review in Section 6.

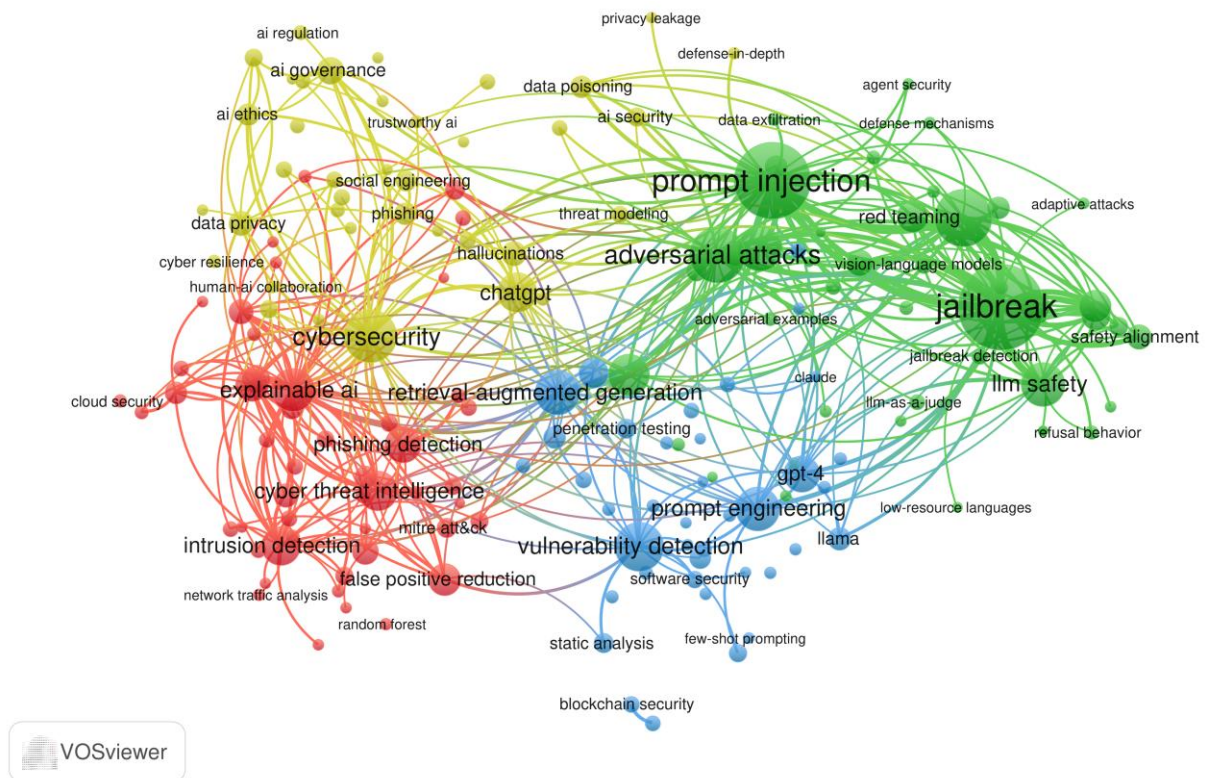


Figure 5. VOSviewer network visualisation of keyword co-occurrence (170 keywords with ≥ 40 occurrences; association-strength normalisation; resolution 1.0; the 400 strongest links are shown). Node size is proportional to occurrences. Red: Cluster 1, LLM-enabled detection and security operations; green: Cluster 2, attacks on and alignment of LLMs; blue: Cluster 3, code and vulnerability analysis and LLM engineering methods; yellow: Cluster 4, socio-technical threats and governance.

Cluster 1, LLM-enabled detection and security operations (red; 49 keywords), is anchored by explainable AI (399 occurrences), cyber threat intelligence (354), intrusion detection (341), phishing detection (309), anomaly detection (257), false-positive reduction (245), malware detection (176), incident response and the security operations centre (129 each). It corresponds to the shield role applied to networks, messaging and operations; 57.8% of its total link strength is internal and only 11.2% connects to the attack cluster. Cluster 2, attacks on and alignment of LLMs (green; 44 keywords), is anchored by jailbreak (1,263), prompt injection (1,050), LLM security (653), adversarial attacks (622) and LLM agents (425), with satellite terms for safety alignment, red teaming, black-box attacks, multimodal models and indirect prompt injection. It corresponds to the target role and is the most internally cohesive cluster: 63.8% of its link strength is internal and only 10.0% connects to the detection cluster. Cluster 3, code and vulnerability analysis (blue; 39 keywords), joins vulnerability detection, static analysis, software and smart-contract security and code generation with LLM engineering methods (prompt engineering, retrieval-augmented generation, fine-tuning and multi-agent systems) and with penetration testing. It is the least self-contained cluster (37.7% internal) and the one most strongly linked to the attack cluster (27.9%), reflecting its position as the methodological bridge of the field. Cluster 4, socio-technical threats and governance (yellow; 38 keywords), contains cybersecurity as a general term, ChatGPT, AI governance, ethics, privacy and regulation together with phishing, social engineering, deepfakes, malware generation and data poisoning, and has the highest mean citations per document (8.3).

Table 8. Thematic clusters of the keyword co-occurrence network (VOSviewer).

Cluster	Label	Keywords	Leading keywords (occurrences)	Internal link share	Dominant role
1 (red)	LLM-enabled detection and security operations	49	explainable AI (399); cyber threat intelligence (354); intrusion detection (341); phishing detection (309); anomaly detection (257); false-positive reduction (245); malware detection (176)	57.8%	Shield (networks and operations)
2 (green)	Attacks on and alignment of LLMs	44	jailbreak (1,263); prompt injection (1,050); LLM security (653); adversarial attacks (622); LLM agents (425); LLM safety (420); red teaming (239)	63.8%	Target
3 (blue)	Code and vulnerability analysis; LLM engineering methods	39	vulnerability detection (499); retrieval-augmented generation (434); prompt engineering (420); multi-agent systems (212); penetration testing (109); static analysis (106)	37.7%	Shield (software) and weapon
4 (yellow)	Socio-technical threats and governance	38	cybersecurity (570); ChatGPT (350); AI governance (187); data privacy (159); phishing (133); data poisoning (132); social engineering (113)	42.5%	Weapon (human-targeted) and governance

Three features of the map bear directly on the framework. First, penetration testing is located in the code-and-vulnerability cluster rather than in a separate offensive cluster: technical offence is studied with the same vocabulary, methods and data as vulnerability detection, consistent with proposition C2. Second, human-targeted offence behaves differently. Phishing, social engineering, deepfakes and malware generation sit in the socio-technical cluster with governance and ethics, whereas phishing detection belongs to the detection cluster. Offensive uses aimed at people are therefore discussed predominantly as societal risks, and their technical counter-measures are studied by a different community. Third, the separation between the attack cluster and the detection cluster is the sharpest in the map. Agent security is discussed within the attack cluster (LLM agents co-occur with prompt injection in 157 documents), whereas agentic AI for defence sits within the detection cluster; only 11 documents link LLM agents to the security operations centre. The research community that builds defensive agents and the community that attacks agents are, to a large extent, different communities.

5.3 Temporal evolution: overlay analysis (RQ2)

Figure 6 colours each keyword by the mean publication date of the documents that contain it. Because the corpus is young, differences are compressed, but the ordering is informative. Terms associated with the first wave of research ChatGPT, GPT-4, BERT, fine-tuning, harmful content generation and AI ethics have the earliest mean dates, while agentic and systems-security terms are the most recent. Table 9 quantifies this shift by comparing the share of documents containing each keyword in 2024 and 2026.

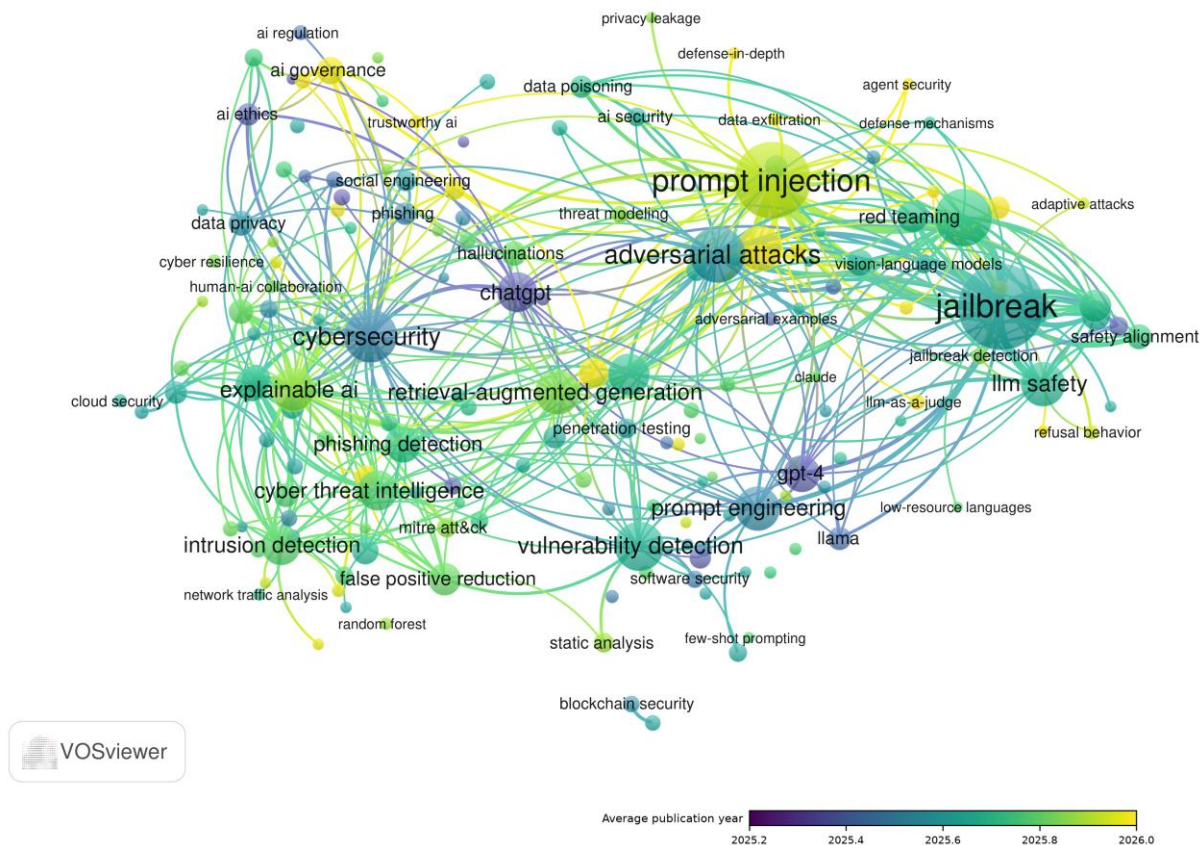


Figure 6. VOSviewer overlay visualisation of the keyword network. Colour encodes the mean publication date of the documents containing each keyword (scale 2025.2–2026.0; yellow = more recent).

Table 9. Emerging and declining keywords (share of documents in each year containing the keyword).

Keyword	2024 (%)	2026 (%)	Direction
Model Context Protocol	0.00	1.73	Emerging
Defense-in-depth	0.00	1.09	Emerging
Agent security	0.08	1.06	Emerging
Prompt injection defense	0.17	1.33	Emerging
Data exfiltration	0.17	1.09	Emerging
Zero-trust architecture	0.17	1.06	Emerging
Indirect prompt injection	0.50	3.00	Emerging
Multi-agent systems	0.83	4.15	Emerging
AI governance	0.83	3.69	Emerging
Security operations centre	0.67	2.94	Emerging
ChatGPT	11.10	1.24	Declining
GPT-4	7.26	2.00	Declining
Harmful content generation	3.17	0.88	Declining
Fine-tuning	3.01	0.70	Declining
BERT	1.75	0.42	Declining

The trajectory is clear. The field has moved from studying what a single chatbot can be induced to say (harmful content, model-specific jailbreaks) towards studying what an LLM-based system can be induced to do: exfiltrate data, misuse tools, propagate through multi-agent workflows and exploit protocol-level integration such as the Model Context Protocol. The simultaneous rise of defence-in-depth, zero-trust

architecture and prompt-injection defence indicates a shift from model-centric to architecture-centric defence. In terms of the three-layer model, research attention is moving upwards from the model layer to the system and agentic layers precisely the layers for which, as Section 6.5 shows, evaluation infrastructure is weakest.

5.4 Intellectual structure: co-citation analysis (RQ2)

The co-citation network of the 95 most co-cited references (after removal of corrupted records) resolves into three clusters in VOSviewer (Figure 7; Table 10). The first is the foundation of LLMs and adversarial prompting. It combines core works of language modelling such as GPT-3 (Brown et al., 2020) with the seminal attack literature: indirect prompt injection (Greshake et al., 2023), in-the-wild and automated jailbreaks (Wei et al., 2023; Zou et al., 2023; Deng et al., 2024; Shen et al., 2024; Chao et al., 2025), privacy jailbreaks (Li et al., 2023) and early defences such as self-reminders (Xie et al., 2023). The second is LLMs for cybersecurity. It contains the broad surveys (Gupta et al., 2023; Yao et al., 2024; Das et al., 2025; Ferrag et al., 2025; Xu et al., 2025; Zhang et al., 2025), intrusion-detection datasets (Moustafa and Slay, 2015; Sharafaldin et al., 2018), threat-intelligence knowledge graphs (Hu et al., 2024) and, importantly, the main offensive studies, including LLM-assisted penetration testing (Happe and Cito, 2023) and the generation and detection of phishing emails (Heiding et al., 2024). The third is code vulnerability detection. It draws on pre-LLM deep-learning vulnerability detection (Chakraborty et al., 2022) and on critical evaluations of LLM-based detection (Ullah et al., 2024; Ding et al., 2025).

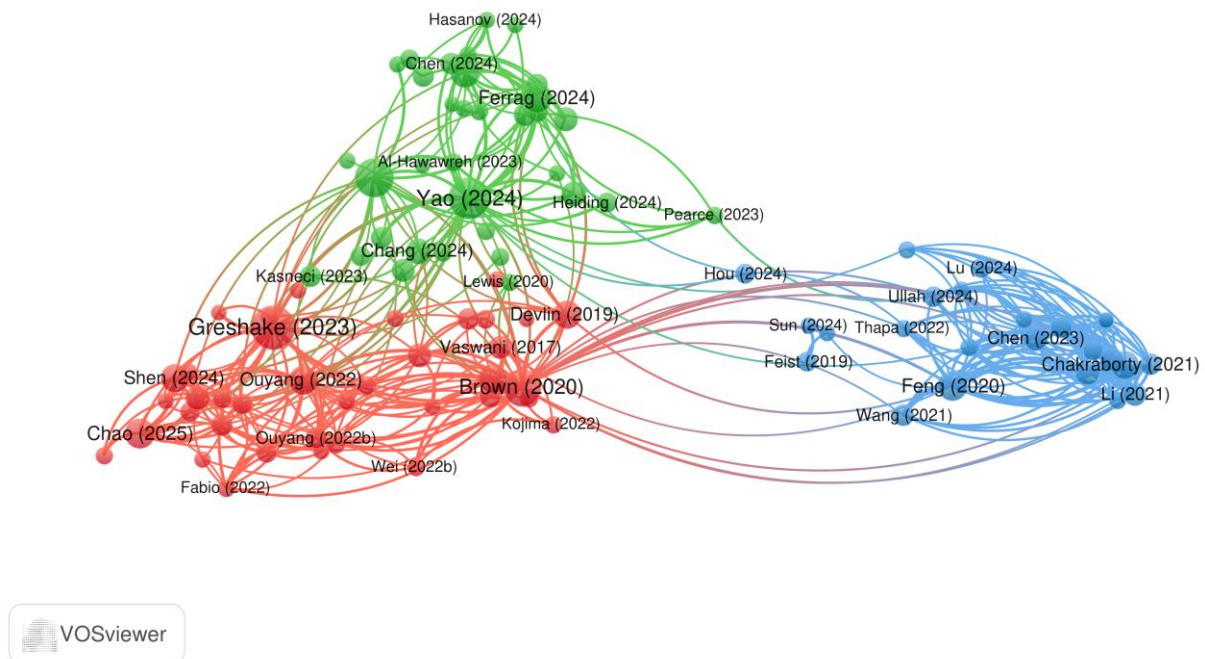


Figure 7. VOSviewer network visualisation of the co-citation network of the 95 most co-cited references (association strength; resolution 1.0). Red: foundations of LLMs and adversarial prompting; green: LLMs for cybersecurity; blue: code vulnerability detection. Node size is proportional to the number of citing documents; labels show first author and year.

Table 10. Intellectual base: co-citation clusters (VOSviewer).

Cluster	References	Most co-cited works (citing documents)	Interpretation
1 (red). Foundations of LLMs and adversarial prompting	38	Greshake et al. 2023 (164); Brown et al. 2020 (124); Chao et al. 2025 (86); Shen et al. 2024 (77); Wei et al. 2023 (61); Zou et al. 2023 (45); Deng et al. 2024 (42)	The target role draws on the NLP and machine-learning canon
2 (green). LLMs for cybersecurity	34	Yao et al. 2024 (146); Gupta et al. 2023 (131); Ferrag et al. 2024 (86); Zhang et al. 2025 (83); Happe and Cito 2023 (61); Sharafaldin et al. 2018 (47); Heiding et al. 2024 (42)	The shield and technical weapon roles share one knowledge base

3 (blue). Code vulnerability detection	23	Feng et al. 2020 (85); Chakraborty et al. 2022 (75); Fan et al. 2020 (59); Chen et al. 2023 (57); Ullah et al. 2024 (45); Ding et al. 2025 (29)	Software-security lineage predating LLMs
--	----	---	--

The co-citation structure supports and refines the co-word results. The technical weapon role does not have its own intellectual foundation base: offensive studies are co-cited with defensive studies and datasets, as proposed in C2. Unlike the defensive applications, the target role is based on an alternative canon from the NLP and the adversarial machine learning literature. This is the intellectual divide between the first two clusters, the "signature" as it were, of the gap that proposition C1 points to: the researchers that build defensive LLM systems read and cite different literatures and so do the researchers that attack LLM systems.

5.5 Dual-use coupling (RQ3)

The direct test of the framework is in the role classification. Of the 7,369 documents, 1,631 (22.1%) do not correspond to any of the dictionaries, generally because they are about LLM security at a general or policy level. The roles are mostly analyzed separately (Figure 8a). Only 7.9% (584) of documents focus on two or more roles; just 0.5% (40) address all three roles. Multi-role share has increased, from 3.4% in 2023 to 9.9% in 2026 (Figure 4), although this is still at a low level.

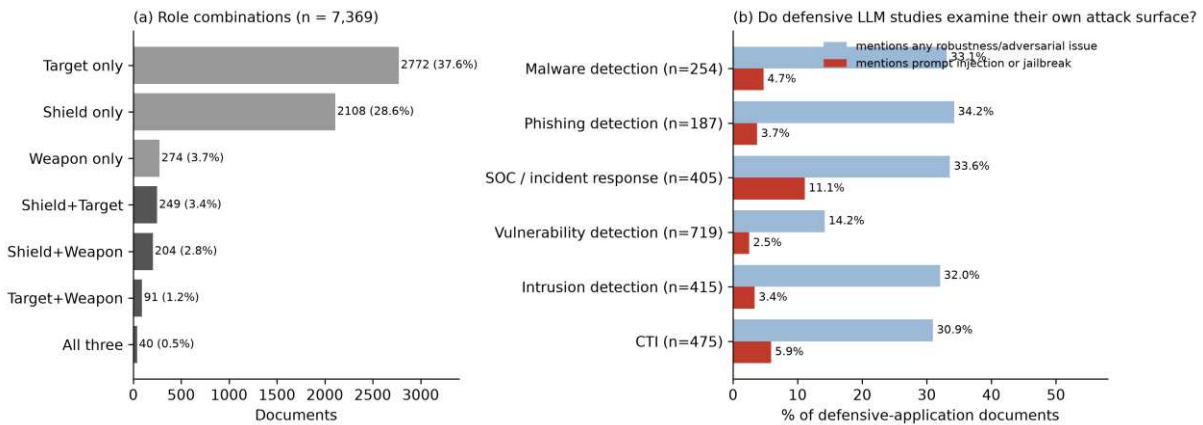


Figure 8. Dual-use coupling. (a) Distribution of documents across role combinations. (b) Share of documents in each defensive application area whose title or abstract refers to prompt injection or jailbreaking (red) or to any robustness, adversarial, poisoning or evasion issue (blue).

This asymmetry that is predicted by proposition C1 is more distinct (Figure 8b). Only 289 defensive documents (11.1%) also address attacks on LLMs. The percentages of documents containing each of the two application areas with the highest potential to negatively impact a deployed defensive model, prompt injection and jailbreaking, range from 2.5% in vulnerability detection, through 3.4% in intrusion detection, to 5.9% in threat intelligence and 11.1% in SOC and incident-response automation applications. These comments about robustness or adversarial concerns are somewhat around the classical evasion of the detector's input features, and not much about the manipulation of the LLM via the content it processes. About one third of defensive documents contain some reference to robustness or adversarial factors, but mostly it's classical avoidance of the detector's input features rather than the manipulation of the LLM through the content it processes. The keyword co-occurrence counts also paint a picture: cyber threat intelligence appears in just 15 of 354 documents, intrusion detection in 9 of 499 and vulnerability detection in 18 of 341.

Another asymmetry is in the citations. Normalized citations for defensive, target, and weapon documents are respectively 1.15, 0.79 and 0.71 times the year average. Thus, with a high number of documents published, the large and rapidly growing target literature receives comparatively little attention per document, a pattern typical of a preprint-driven literature.

Last but not least, the abstract-level indicators of the nature of evaluation practice are uniformly scarce. Among target-role documents, 3.2% mention adaptive attacks, 1.8% mention repeated runs, variance or statistical significance, 3.3% mention multilingual or cross-lingual evaluation and 1.3% mention longitudinal or cross-version evaluation; 13.9% mention utility or false-positive costs. Across the whole corpus, only 2.0% report a human-subject component. These indicators measure what abstracts report rather than what studies do, and therefore underestimate practice; they nonetheless identify which evaluation properties authors consider worth foregrounding, and they anticipate the evidential problems documented in Section 6.

6. Critical literature analysis (RQ4)

This section examines the evidence within each role, following the clusters identified in Section 5: the attack cluster (Section 6.1), the socio-technical and code clusters for offensive use (6.2), the detection and code clusters for defensive use (6.3), and the defences and evaluation practices that cut across all clusters (6.4 and 6.5). Throughout, we report results as stated by the primary studies together with the qualifiers that determine how far they generalise.

6.1 LLMs as targets

6.1.1 The model layer

Model-layer threats target model weights or the alignment procedure independently of any surrounding application. Jailbreaking manipulating a prompt so that a safety-aligned model produces output its policy forbids is the largest topic in the corpus (1,263 keyword occurrences), and its technique space is now well mapped (Wei et al., 2023; Yi et al., 2024). Wei et al. (2023) provide the most influential explanation of why safety training fails, attributing jailbreaks to competing objectives and mismatched generalisation between capability and safety training. Shen et al. (2024) analysed 1,405 in-the-wild jailbreak prompts collected over thirteen months, identified 131 jailbreak communities and found that the most effective prompts achieved attack success rates of 0.95 against GPT-3.5 and GPT-4 and remained online for more than 240 days.

Automation followed in two directions. Optimisation-based attacks search for adversarial suffixes: Zou et al. (2023) showed that suffixes found on open-weight surrogates transfer to black-box commercial systems, although their unreadable form makes them detectable by perplexity filters. Instead, Semantic attacks create fluent prompts using a hierarchical genetic algorithm (AutoDAN, Liu et al., 2023a) or an attacker LLM to tweak prompts only twenty times (PAIR, Chao et al., 2025) or a timing-based approach adapted from SQL injection (MasterKey, Deng et al., 2024) to deduce the provider defences, before they are used to create the attacks. Variants of persuasion, multi-turn and multimodal further broaden the space (Yi et al., 2024).

Training-time attacks, such as data poisoning and backdoor insertion, have access to the training pipeline and are less studied empirically (Das et al., 2025), but are the most serious class due to the fact that a poisoned model will be vulnerable to such attacks in every deployment. Das et al. (2025) and Pribisalić and Martinčić-Ipšić (2026) discuss privacy attacks in detail, including membership inference, training-data extraction, and the leakage of personally identifiable information, and highlight that such attacks can be triggered through multi-step jailbreaks (Li et al., 2023) due to probabilistic generation and large-scale data ingestion, rather than simply implementation flaws.

6.1.2 The system layer

Application-layer threats attack the application that is based on the model. The model's weakness is that trusted instructions and data are interleaved in a single context window, and the model lacks a secure way to differentiate between the two. Demos in the early stages revealed that simple “ignore previous instructions” prompts can be used to hijack model behaviour (Perez and Ribeiro, 2022). The most locally cited document in our corpus, Greshake et al. (2023), defined a new threat category, indirect prompt injection, where the attacker does not have to be in front of the user, but simply hiding in the content that

the application fetches, e.g., a web page, an email, a document, or a code comment. This makes it possible for indirect injection to scale to everyone who can publish content that a target system could read. Liu et al. (2023b) provided the first formal framework showing that previously reported attacks are special cases of a common structure. The BIPIA benchmark (Yi et al., 2025) concluded that LLMs are universally vulnerable to indirect injection because they cannot distinguish information from instructions and have no built-in disposition to refuse instructions from external sources.

Retrieval-augmented generation introduces knowledge-base poisoning, which attacks the corpus rather than the model; Gulyamov et al. (2026) report research in which as few as five crafted documents manipulated responses in 90% of cases. Multimodal injection embeds instructions in images or audio; an evaluation of eight commercial models found exploitable weaknesses in all of them in the absence of supplementary sanitisation (Yeo and Choi, 2025).

6.1.3 The agentic layer

Agentic systems add persistent memory, tool invocation, multi-step planning and inter-agent communication, each of which creates threats that do not exist for a stateless model (He et al., 2025). The overlay analysis shows that this layer is where research attention is moving fastest (Table 9). Ferrag et al. (2026) present a unified threat model for agent ecosystems covering host-to-tool and agent-to-agent communication, categorise more than thirty attack techniques and validate the framework against real incidents recorded in CVE and the NIST National Vulnerability Database, including prompt-to-SQL injection and exploits against GitHub MCP servers. Gulyamov et al. (2026) document a remote-code-execution vulnerability in GitHub Copilot (CVE-2025-53773, CVSS 9.6). Narajala and Narayan (2025) organise agent-specific risk into five domains cognitive architecture, temporal persistence, operational execution, trust-boundary violation and governance circumvention and emphasise delayed exploitability and cross-system propagation as failure modes that conventional threat models do not capture. Benchmarks for tool-integrated agents confirm the practical severity of indirect injection in this setting (Zhan et al., 2024).

Chu (2026) provides the most structurally useful framing. The layered attack-surface model decomposes the agentic stack into seven layers (foundation, cognitive, memory, tool execution, multi-agent coordination, ecosystem and governance) and adds a temporal axis with four classes (instantaneous, session-persistent, cross-session cumulative and sub-session-stack). Applying this framework to 116 papers, Chu finds that the upper layers are sharply under-explored, that several documented attack regions have no corresponding defences and that no current benchmark covers cross-session or sub-session-stack failure modes. Su et al. (2025) reach a compatible conclusion from the perspective of autonomy, tracing memory poisoning, tool misuse, reward hacking and emergent misalignment to architectural fragilities in perception, cognition, memory and action modules. Deng et al. (2026) apply a five-stage lifecycle framework to a widely deployed autonomous agent and demonstrate compound threats, including skill supply-chain contamination and intent drift, concluding that point-based defences fail against cross-temporal risks.

Empirically, agentic systems are markedly easier to attack than the models inside them. The Agent Security Bench (Zhang et al., 2024a) tests ten prompt-injection attacks, memory poisoning, a Plan-of-Thought backdoor and four mixed attacks on thirteen LLM back-ends, reporting a highest average attack success rate of 84.30% and limited effectiveness of the eleven defences tested. Malfunction-amplification attacks induce repetitive or irrelevant actions rather than harmful outputs and evade LLM self-examination, with failure rates above 80% (Zhang et al., 2024b). Reviews of the resulting defence landscape converge on two recurring weaknesses: poor generalisation of defences under attacker adaptation and limited assessment of agent-tool interaction risk (Tamuka et al., 2026). Table 11 consolidates the three layers into a taxonomy that is used for the remainder of the article.

Table 11. Three-layer taxonomy of attacks on LLM-based systems.

Layer	Threat class	Mechanism	Persistence	Representative works
Model	Jailbreak (manual and in-the-wild)	Adversarial prompt bypasses alignment	Single turn	Wei et al., 2023; Shen et al., 2024
Model	Jailbreak (optimisation-based)	Automated adversarial-suffix search	Single turn	Zou et al., 2023
Model	Jailbreak (semantic, automated)	Genetic, fuzzing or LLM-generated prompts	Single turn	Liu et al., 2023a; Deng et al., 2024; Chao et al., 2025
Model	Data poisoning and backdoors	Corrupted training or fine-tuning data	Permanent	Pribisalić and Martinčić-Ipšić, 2026
Model	Privacy extraction	Membership inference, PII leakage	Not applicable	Li et al., 2023; Das et al., 2025; Pribisalić and Martinčić-Ipšić, 2026
System	Direct prompt injection	Malicious instruction in user input	Single turn	Perez and Ribeiro, 2022; Liu et al., 2023b
System	Indirect prompt injection	Instruction in retrieved web page, email or document	Single turn	Greshake et al., 2023; Yi et al., 2025
System	Multimodal injection	Instruction in image or audio channel	Single turn	Yeo and Choi, 2025
System	RAG and knowledge-base poisoning	Crafted documents in retrieval corpus	Persistent	Gulyamov et al., 2026
Agentic	Memory poisoning	Corrupted persistent memory	Cross-session	Zhang et al., 2024a; Su et al., 2025
Agentic	Tool and plugin exploitation	Abuse of the function-calling interface	Session	Zhan et al., 2024; Narajala and Narayan, 2025; Ferrag et al., 2026
Agentic	Protocol-level exploits (MCP)	Tool poisoning, credential theft	Persistent	Ferrag et al., 2026; Gulyamov et al., 2026
Agentic	Supply-chain and skill contamination	Malicious agent skills or plugins	Persistent	Deng et al., 2026
Agentic	Malfunction amplification	Induced repetitive or irrelevant action loops	Session	Zhang et al., 2024b
Agentic	Intent drift and goal misalignment	Gradual divergence over long horizons	Cross-session	Su et al., 2025; Deng et al., 2026

6.2 LLMs as weapons

6.2.1 Social engineering, phishing and deepfakes

Social engineering is the most immediate offensive application because it requires little technical sophistication and targets the component that defences find hardest to protect: the human. Phishing, social engineering and deepfakes are part of the socio-technical cluster while phishing detection constitutes the detection cluster are examples of attack-side terms. According to Schmitt and Flechais (2024), generative AI can be used to enhance social engineering efforts in three ways realistic content creation, advanced targeting and personalisation, and automated attack infrastructure. Gonzaga et al. (2026) make a similar trio, taking the analysis from description to economics by formulating the attacker's optimal strategy as a Markov decision process.

Three empirical findings are emphasized. In one study, even Heiding et al., (2024) one of the most locally cited offensive studies, compares LLM created phishing emails to human-created phishing emails and assesses LLMs at detecting phishing emails, showing in one study, that the generative and detective aspects of the phishing capability are woven into one. Second, Mishra and Varshney (2025) present controlled experiments that indicate a de-skilling effect, as previously claimed, in which novice subjects' perceived competence to spot phishing attacks rose by 240% and task completion time dropped by 400% compared to traditional web resources, with the help of a jailbroken version of a commercial system. Third, AI-generated phishing emails remain stylistically distinguishable from human-written scam emails and can be detected with high accuracy by classifiers trained on AI-generated corpora (Eze and Shamir, 2024), so the de-skilling effect is at least partially offset by a new detection signal, provided that detectors are retrained. A systematic review of human factors concludes that technical detection alone is insufficient because these attacks exploit

psychological rather than technical vulnerabilities (Jabir et al., 2025). Purpose-built malicious models such as FraudGPT and WormGPT remove the jailbreaking step entirely (Ferrara, 2024), although their real-world capability relative to jailbroken frontier models remains poorly characterised in the peer-reviewed literature. Ferrara (2024) surveys the wider space of nefarious applications, including synthetic identities and coordinated influence operations.

6.2.2 Malware generation

There is less primary evidence of LLM created malware than there is commentary about it. The evolution of polymorphic malware and signature-based evasion have been covered by Ferrara (2024), and the automated generation of payloads and hacking have been included as misuses of generative AI by Gupta et al. (2023). Controlled evaluations are very unusual for sound ethical and legal considerations. This is an area where review evidence often outstrips primary evidence, so we are cautious about treating capability claims as findings.

6.2.3 Autonomous exploitation and penetration testing

This module covers autonomous exploitation and penetration testing. This module is on autonomous exploitation and penetration testing.

This is the region where most daring offensive assertions are made, and where most conflicted results are found. It is part of the code and vulnerability group in the bibliometric map and in the co-citation network it co-cites with defensive surveys (consistent with proposition C2). Happe and Cito (2023) conducted a preliminary study of GPT-3.5 as an assistant in penetration testing. Happe and Cito (2023) did a preliminary study on the use of GPT-3.5 as a penetration test assistant. They then benchmarked it against GPT-4-Turbo, finding that it could exploit anywhere from 33% to 83% of the Linux privilege-escalation vulnerabilities they tried, depending on the given guidance and context management, while the human professional baseline was at 75% and was also on par with the cost of exploitations.

Fang et al. produced two of the most cited results in this line: LLM agents can autonomously hack websites, including through blind SQL injection, without prior knowledge of the vulnerability (Fang et al., 2024b), and GPT-4 can exploit 87% of a set of 15 one-day vulnerabilities when given the CVE description a rate that falls to 7% without it (Fang et al., 2024a). The second figure is the single most important qualifier in the offensive literature and is frequently omitted when the first is quoted. RapidPen (Nakatani, 2025) demonstrates fully automated IP-to-shell compromise at roughly USD 0.30–0.60 and 200–400 seconds per run.

More recent benchmarks are considerably more sober. CVE-Bench (Zhu et al., 2025), using real critical-severity CVEs in sandboxed web applications, found state-of-the-art agent frameworks exploiting up to 13%. CyberGym (Wang et al., 2025), spanning 1,507 real vulnerabilities across 188 projects, found the best combinations achieving roughly 20%, although the benchmarking process itself discovered 34 zero-day vulnerabilities and 18 historically incomplete patches a clear instance of proposition C2. PACEbench (Liu et al., 2025b), designed with realistic difficulty, environmental complexity and active defences, found that no evaluated frontier model could bypass the defences and concluded that current models do not yet pose a generalised cyber-offence threat. PentestEval (Yang et al., 2025) reports end-to-end pipeline success of 31%, with fully autonomous agents failing almost entirely, and Luo et al. (2026) report success rates from 10.7% to 69.3% across 19 models on 300 target servers, tracking general model capability.

Two studies help reconcile these numbers. Yu et al. (2026) show that agents achieve up to 90% functional exploitation success when given accurate vulnerability information but only around 50% recall when performing reconnaissance without assistance, largely because they fail to process unstructured telemetry. End-to-end benchmarks that do not separate these stages therefore conflate errors: they understate exploitation ability while overstating autonomy. Erdem (2026) provides the methodologically most important result: 400 runs of four models against a single fixed target produced exploitation rates between

25% and 85%, with statistically significant differences between models and distinct failure modes; the study also documents its own correction of an earlier draft in which upstream API errors had been recorded as safety refusals. Single-run results in this literature should be read in that light. Table 12 and Figure 9 summarise the reported capabilities.

Table 12. Reported offensive capability of LLM agents.

Study	Task	Target	Reported result	Key qualifier
Happe et al., 2026	Linux privilege escalation	Curated single-vulnerability VMs	GPT-4-Turbo 33–83%	Human baseline 75%; guidance strongly affects success
Fang et al., 2024b	Autonomous website hacking	Sandboxed web applications	GPT-4 succeeds; open models largely fail	Frontier-model capability at the time
Fang et al., 2024a	One-day vulnerability exploitation	15 real CVEs	87% with CVE description	7% without the description
Nakatani, 2025	IP to shell	Hack The Box target	60% with reuse of prior successes; USD 0.3–0.6 per run	Single target
Zhu et al., 2025	Real CVE exploitation	40 critical CVEs, sandboxed	Up to 13%	Realistic setting
Wang et al., 2025	Proof-of-concept generation	1,507 vulnerabilities, 188 projects	~20% (best)	Also found 34 zero-days
Liu et al., 2025b	Chained and defended exploitation	Purpose-built scenarios	No model bypassed defences	Defences are the differentiator
Yang et al., 2025	Stage-decomposed penetration test	12 scenarios, 346 tasks	31% end-to-end	Fully autonomous agents nearly fail
Luo et al., 2026	Autonomous penetration	300 servers, two tiers	10.7–69.3% across 19 models	Scales with general capability
Yu et al., 2026	Reconnaissance vs. exploitation	70 web testbeds	90% exploitation with context; ~50% reconnaissance recall	Locates the bottleneck
Erdem, 2026	Consistency under repetition	One target, 400 runs	25–85% by model	Variance, not point estimates, is the finding

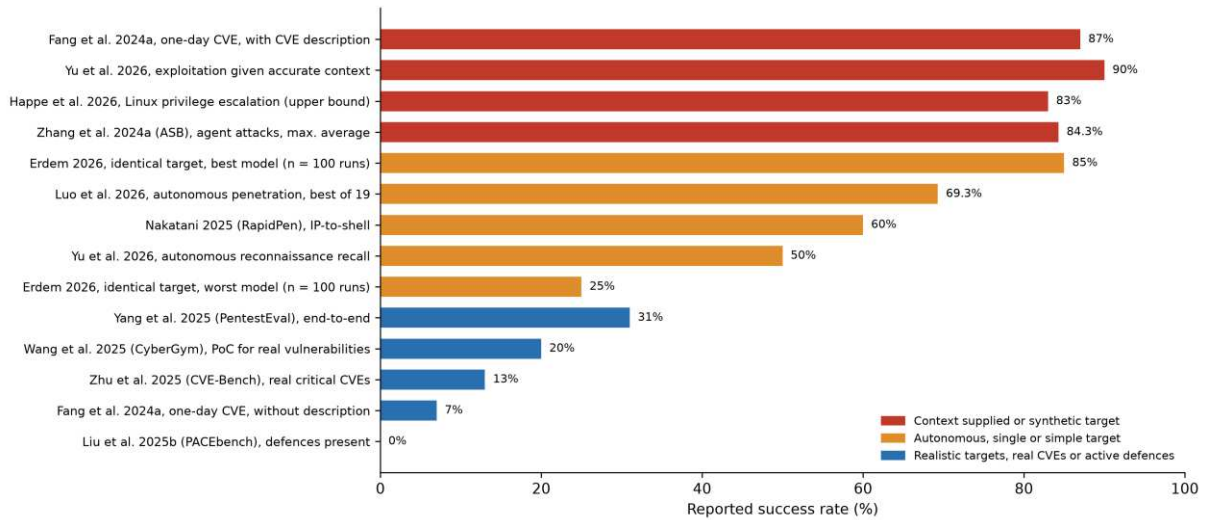


Figure 9. Reported success rates of offensive LLM agents, drawn from Table 12. The spread tracks experimental design more closely than model capability: results are high when vulnerability context is supplied or targets are synthetic, and low when targets are real CVEs or defences are present.

6.3 LLMs as shields

6.3.1 Cyber threat intelligence

Threat intelligence is the natural first defensive application because its raw material is unstructured text at a volume no analyst team can read (Chen et al., 2024). LLM-TIKG (Hu et al., 2024) constructs threat-intelligence knowledge graphs from open-source reporting, using few-shot GPT annotation to bootstrap a

fine-tuned 7B model and reporting 87.88% precision for named-entity recognition and 96.53% for TTP classification. CyLens (Liu et al., 2025a) incorporates 271,570 threat reports into an agentic copilot that supports attribution, contextualisation, correlation, prioritisation and remediation. The indispensable counterweight is Mezzi et al. (2025), who evaluated three state-of-the-art models on 350 real threat-intelligence reports in zero-shot, few-shot and fine-tuned settings, explicitly measuring consistency and confidence calibration. The models could not sustain adequate performance on full-length reports and were both inconsistent and overconfident; few-shot learning and fine-tuning offered only partial improvement. In the light of proposition C1, a further observation is essential: threat-intelligence pipelines ingest attacker-controllable text by design, yet only 5.9% of threat-intelligence documents in our corpus mention prompt injection or jailbreaking.

6.3.2 Intrusion and threat detection

SecurityBERT (Ferrag et al., 2024) applies BERT with a privacy-preserving fixed-length encoding to IoT and IIoT traffic, reporting 98.2% accuracy across fourteen attack types with a 16.7 MB model and inference below 0.15 seconds on a CPU, which makes edge deployment plausible. A modular multi-agent architecture with specialised email, log and IP agents and a contextual correlation layer reports 93.6% detection accuracy, 87% correlation accuracy and a 41.3% reduction in false positives compared with traditional approaches (Hmimou et al., 2025). PLLM-CS, which extends the approach to satellite networks, reports 100% accuracy on UNSW-NB15 (Hassanin et al., 2024).

These accuracy figures require caution. Results at or near 100% on NSL-KDD, UNSW-NB15 and CIC-IDS2017 are common across the deep-learning intrusion-detection literature and largely reflect known limitations of these datasets redundancy, labelling errors, artefacts of traffic generation and the gap between closed-world benchmarks and operational traffic rather than deployable capability (Tavallaei et al., 2009; Sommer and Paxson, 2010; Arp et al., 2022). The co-citation analysis shows that two of these datasets (Moustafa and Slay, 2015; Sharafaldin et al., 2018) remain part of the intellectual base of the field, which suggests that the problem is being inherited rather than corrected.

6.3.3 Vulnerability detection and secure code

According to the third co-citation cluster of the pre-LLM lineage (Figure 1), vulnerability detection is the largest defensive topic (499 keyword occurrences) and the most ancient one. The third co-citation cluster of the pre-LLM lineage is Vulnerability detection, which is the largest defensive topic (499 occurrences) and the most ancient one (Chakraborty et al., 2022). A remarkable individual result is IRIS (Li et al., 2024), which integrates LLM-inferred taint specifications and static analysis to conduct whole-repository reasoning. CWE-Bench-Java was used to evaluate the tools, and on this benchmark, CodeQL detected 27 of 120 validated vulnerabilities, whereas IRIS with GPT-4 detected 55 and found four vulnerabilities that were not detected by any existing tool. The benefit of increased recall is an increase in false positives, a trade-off that is common for all of them. However, cautious conclusions temper the optimism: Ullah et al. (2024) report that LLMs are not yet capable of finding and reasoning about security vulnerabilities, and the PrimeVul evaluation reveals that the performance of these models on realistic de-duplicated data is significantly lower than they claim with respect to previous tests (Ding et al., 2025). However, Zero-shot repair is promising but inconsistent (Pearce et al., 2023). Secure-generation techniques reinforcement learning with security rewards, retrieval of CWE-specific guidance, prompt optimisation and multi-agent analysis with fuzzing have attracted parallel effort, but their evaluation is contested, as discussed below.

The mirror image of this work is the security of LLM-generated code. Early evidence showed that code assistants frequently suggest vulnerable code (Pearce et al., 2022). CWEval (Peng et al., 2025) argues that evaluating functionality and security separately misreports both. The most important negative result is that of Dai et al. (2025), who evaluated four secure-code-generation techniques with security and functionality assessed jointly on the same output. These techniques often achieved security by degrading functionality in

many cases by more than 50% relative to the base model frequently by deleting vulnerable lines or emitting unrelated code; CodeQL, the analyser on which much of this literature relies, also missed several vulnerabilities.

6.3.4 Security operations and response

Xu et al. (2025) identify a shift from single-task execution towards LLM-based agents that orchestrate multi-step security workflows and note that training and evaluation datasets remain limited. Broad surveys (Ferrag et al., 2025) cover reporting, incident-response planning, threat hunting and secure code generation and anticipate self-healing systems and autonomous cyber-defence agents. The overlay analysis shows that SOC automation is among the fastest-growing defensive topics (Table 9). SOC and incident-response documents also show the highest engagement with prompt injection among defensive areas (11.1%), but this still means that almost nine in ten such studies do not foreground the manipulation of the agent through the logs, tickets and reports it processes. Table 13 summarises the defensive applications.

Table 13. Defensive applications of LLMs in cybersecurity.

Area	Task	System	Source	Reported performance	Principal limitation
Threat intelligence	Knowledge-graph construction	LLM-TIKG	Hu et al., 2024	NER precision 87.88%; TTP 96.53%	Domain-specific fine-tuning required
Threat intelligence	Full-lifecycle copilot	CyLens	Liu et al., 2025a	Outperforms baseline LLMs and agents	Not independently replicated
Threat intelligence	Reliability assessment	Negative result	Mezzi et al., 2025	Inconsistent and overconfident	Fails on full-length reports
Intrusion detection	IoT/IloT traffic classification	SecurityBERT	Ferrag et al., 2024	98.2% over 14 attack types; 16.7 MB	Benchmark dataset limitations
Intrusion detection	Satellite-network IDS	PLLM-CS	Hassanin et al., 2024	100% on UNSW-NB15	Saturated benchmark
Intrusion detection	Multi-source correlation	Multi-agent architecture	Hmimou et al., 2025	93.6% detection; 41.3% FP reduction	Partly simulated environment
Vulnerability detection	Whole-repository analysis	IRIS	Li et al., 2024	55/120 vs. CodeQL 27/120; 4 novel	Java only
Vulnerability detection	Reasoning about vulnerabilities	Critical evaluation	Ullah et al., 2024; Ding et al., 2025	Unreliable identification and reasoning	Performance drops on realistic data
Secure code generation	Joint security–functionality evaluation	Negative result	Dai et al., 2025	Security gains often destroy functionality	Base model degraded > 50% in cases

6.4 Defence mechanisms for LLM systems

Defences for LLM systems fall into four families. Correia et al. (2026) catalogue 88 studies against an extension of the NIST adversarial-machine-learning taxonomy (Vassilev et al., 2024), and we adopt a compatible grouping. Figure 10 places each family at the stage of the processing pipeline it occupies and pairs it with a published result that bypasses it; Table 14 summarises the families.

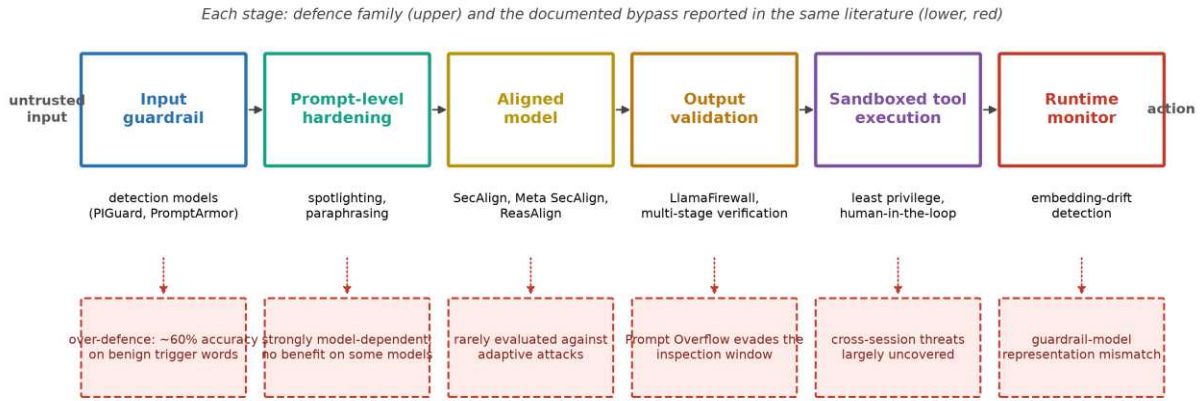


Figure 10. Defence-in-depth pipeline for LLM systems. Each stage shows the defence family that occupies it (upper) and a documented bypass or weakness reported in the same literature (lower).

6.4.1 Prompt-level defences

Prompt-level defences modify the prompt without changing the model: spotlighting, delimiters, boundary awareness, explicit reminders, sandwiching and paraphrasing of retrieved content. Self-reminders were an early and widely co-cited example (Xie et al., 2023). The boundary-awareness and explicit-reminder defences of Yi et al. (2025) provide substantial black-box mitigation, and their white-box variant reduces attack success to near zero while preserving output quality. Pai (2026) evaluated five prompting-based defences across three model families and three deployment domains in 3,510 trials, finding paraphrasing of retrieved content the most consistently effective (reducing camouflaged injection success by 55–84%) and finding effectiveness strongly model-dependent: spotlighting halved attack success on one model and provided no benefit on another. Financial-domain deployments showed the highest residual risk, at 26–33% baseline attack success.

6.4.2 Detection and guardrail models

Detection defences place a classifier in front of the model. The family includes industrial guardrails such as Prompt Guard, Azure Prompt Shield, Granite Guardian and NeMo Guardrails, and research systems such as PIGuard (Li et al., 2025) and PromptArmor (Shi et al., 2025). PromptArmor (Shi et al., 2025), which prompts an off-the-shelf model to detect and remove injected content, reports false-positive and false-negative rates below 1% on AgentDojo, with attack success below 1%, and its authors reasonably propose it as a standard baseline.

Guardrails have three documented structural weaknesses. The first is over-defence: on the NotInject dataset, state-of-the-art guard models fall to near-random accuracy (around 60%) on benign inputs that contain trigger words (Li et al., 2025). The second is evasion: Hackett et al. (2025) achieved up to 100% evasion against six production protection systems using character injection and adversarial machine-learning techniques, and showed that word importance computed offline on white-box surrogates improves attacks on black-box targets. The third is architectural mismatch. The Prompt Overflow attack (Zhou et al., 2026) exploits the gap between a guardrail's limited inspection window and the far larger context window of the model behind it, fragmenting malicious instructions so that no inspected segment appears malicious while the full context remains actionable. Wang et al. (2026) document a related representation asymmetry: guardrails are fragile to character perturbations that the protected model decodes without difficulty, and there is a "success interval" of perturbation strength that defeats detection while preserving the attacker's intent.

6.4.3 Model-level alignment defences

Alignment defences train robustness into the weights. SecAlign (Chen et al., 2025b) uses preference optimisation over paired secure and insecure responses and was among the first methods to reduce prompt-injection success below 10% while generalising to attacks unseen in training. Meta SecAlign (Chen et al., 2025a) released this capability as open weights and evaluated it on nine utility and seven security benchmarks, reporting that security transfers to downstream tasks such as tool calling and web navigation. ReasAlign (Li et al., 2026) adds structured reasoning, a preference-optimised judge and test-time scaling, reporting 94.6% utility at 3.6% attack success on CyberSecEval2, compared with 56.4% utility and 74.4% attack success for Meta SecAlign on the same benchmark. Put differently, the largest reported improvements over each generation of defence come from the authors proposing its successor, and independent replications are scarce.

6.4.4 System-level and architectural defences

Architectural defences assume that the model may be compromised and constrain what it can do: sandboxed execution, least-privilege tool interfaces, human confirmation for irreversible actions, provenance tracking and runtime monitoring. These controls apply long-established security design principles, notably least privilege and complete mediation (Saltzer and Schroeder, 1975), to a new kind of component. LlamaFirewall (Chennabasappa et al., 2025) integrates a jailbreak detector, a chain-of-thought alignment auditor and a static-analysis engine for generated code. Gulyamov et al. (2026) propose PALADIN, a five-layer framework, and characterise prompt injection as a fundamental architectural problem that cannot be fully solved at the model level. From the agent perspective, Narajala et al. (2025) and Deng et al. (2026) stress that controls must be composed across the agent lifecycle rather than applied in isolation. The overlay analysis shows that this architectural turn is the most recent development in the defence literature (Table 9).

Table 14. Defence families for LLM and agentic systems.

Family	Mechanism	Source	Best reported effect	Documented weakness
Prompt-level	Boundary awareness, explicit reminder	Yi et al., 2025	Near-zero ASR (white-box variant)	Strongest form requires model access
Prompt-level	Paraphrasing retrieved content	Pai, 2026	55–84% ASR reduction	Strongly model-dependent
Prompt-level	Spotlighting, sandwiching	Pai, 2026	Halves ASR on some models	No benefit on others
Detection	Off-the-shelf LLM as filter	Shi et al., 2025	< 1% FPR and FNR; ASR < 1%	Cost and latency per call
Detection	Trained guard model	Li et al., 2025	Large gain over prior guards	Over-defence on benign trigger words
Detection	Commercial guardrails	Maiorano, 2026	NeMo 0% bypass	16.22% FPR; up to 100% evasion under AML attack (Hackett et al., 2025)
Model-level	Preference optimisation	Chen et al., 2025b	PI success < 10%	Requires training access
Model-level	Open secure foundation model	Chen et al., 2025a	Security transfers to unseen tasks	Outperformed in successors' reports
Model-level	Reasoning-enhanced alignment	Li et al., 2026	94.6% utility; 3.6% ASR	Single-group evaluation
System-level	Layered guardrail framework	Chennabasappa et al., 2025	Strong jailbreak detection	Alignment auditor experimental
System-level	Defence-in-depth framework	Gulyamov et al., 2026	Conceptual (PALADIN)	Not empirically validated at scale

6.5 Benchmarks and evaluation

Evaluation is where the critical review and the bibliometric indicators converge most directly. Jia et al. (2026) provide the necessary corrective for the defence literature: evaluating existing prompt-injection defences against both existing and adaptive attacks, with diverse target and injected prompts, and measuring retained utility, they conclude that prior studies did not follow a comprehensive methodology and that

existing defences are less effective than reported. This is in line with the findings of the previous 10 years of research on adversarial examples, where defenses tested against non-adaptive attackers were easily defeated after being adapted by the attackers (Athalye et al., 2018; Carlini et al., 2019). In a deployment scenario, Maiorano (2026) calculated the security–utility balance: NeMo Guardrails produced zero false-positive rate at 16.22% bypass, while Prompt Guard delivered a 38.48% bypass rate with a 3.60% false-positive rate. Table 15 is a list of the major benchmarks, followed by the following observations.

First, it's not evenly distributed. Prompt injection and jailbreaking benchmarks are abundant, while those for cross-session, long-horizon and multi-agent threats are practically nonexistent (Chu, 2026). Second, there are no metric labels. Labels like accuracy, F1 and safety score are different ways of expressing the same quantity in different ways in different studies; as such, these labels need to be interpreted as per the study, and should be compared horizontally, rather than vertically, in Tables 12 and 14; many of the reported benchmarks are distinct from the ones catalogued by Solanke (2026), which lists a total of 195. The resulting environment makes real progress difficult to differentiate from selective reporting, argue Cheng et al. (2025). Third, benchmarks decay. Abdelnabi et al. (2026) list three types of failure to evaluate agent-security: English-only benchmarks, temporal staleness and repository staleness, and Solanke (2026) notes that 137 out of 195 benchmarks have stale repositories and 165 are English-only. Fourth, benchmarks are not aligned with regulation. Mapping 194,955 benchmark questions against the EU AI Act taxonomy, Prandi et al. (2025) find that 61.6% of regulation-relevant questions address hallucination and 31.2% performance reliability, while capabilities central to loss-of-control scenarios receive no coverage. Verifiable evaluation using trusted execution environments (Schnabl et al., 2025) addresses part of this problem.

Table 15. Benchmarks and evaluation resources.

Benchmark	Source	Scope	Scale	Notable finding
Open-Prompt-Injection	Liu et al., 2023b	Prompt-injection formalisation	5 attacks, 10 defences, 10 LLMs, 7 tasks	First systematic framework
BIPIA	Yi et al., 2025	Indirect prompt injection	Multi-task	LLMs universally vulnerable
JailbreakHub	Shen et al., 2024	In-the-wild jailbreaks	1,405 prompts; 107,250 questions	0.95 ASR prompts persisting 240+ days
AdvBench / GCG	Zou et al., 2023	Transferable adversarial suffixes	Multi-model	Transfers to closed-source systems
InjecAgent	Zhan et al., 2024	Indirect injection in tool-integrated agents	Multi-tool	Agents broadly vulnerable
Agent Security Bench	Zhang et al., 2024a	Agentic attacks and defences	10 scenarios, 400+ tools, 27 methods	Max. average ASR 84.30%
NotInject	Li et al., 2025	Guardrail over-defence	339 benign samples	Guard models near random ($\approx 60\%$)
CWE-Bench-Java	Li et al., 2024	Vulnerability detection	120 validated vulnerabilities	CodeQL 27; IRIS 55
CWEval	Peng et al., 2025	Joint functionality and security	Multilingual	Prior benchmarks inaccurate
CVE-Bench	Zhu et al., 2025	Real-world CVE exploitation	40 critical CVEs	Best agents $\approx 13\%$
CyberGym	Wang et al., 2025	Vulnerability reproduction	1,507 vulnerabilities, 188 projects	$\approx 20\%$ best; 34 zero-days found
PACEbench	Liu et al., 2025b	Realistic defended exploitation	4 scenario types	No model bypassed defences
PentestEval	Yang et al., 2025	Stage-decomposed penetration testing	346 tasks, 12 scenarios	End-to-end 31%
Bench-2-CoP	Prandi et al., 2025	Benchmark–regulation mapping	194,955 questions	No coverage of loss-of-control capabilities
AlSafetyBenchExplorer	Solanke, 2026	Catalogue of safety benchmarks	195 benchmarks	137 stale repositories; 165 English-only

7. Discussion

7.1 Summary of findings

RQ1 asked how the field has developed. Output grew more than tenfold between 2023 and 2026, is dominated by preprints and conference papers, and is concentrated in the United States and China, with Central Asia, among other regions, almost absent. RQ2 asked about thematic and intellectual structure. The field resolves into four thematic clusters attacks on and alignment of LLMs; detection and security operations; code and vulnerability analysis; and socio-technical threats and governance resting on three intellectual bases: the NLP and adversarial-prompting canon, the LLMs-for-security literature and the software-vulnerability lineage. Research attention is moving from model-level jailbreaks towards system- and agent-level threats and architectural defences. RQ3 asked whether the three roles are studied jointly. They are not: only 7.9% of documents span more than one role, and only 11.1% of defensive studies engage with attacks on the defending model. RQ4 asked how robust the evidence is. Reported offensive success rates for comparable tasks range from 0% to 90% depending on design; defences are seldom evaluated against adaptive attackers; and the costs of defence to legitimate users are rarely reported. The following subsections interpret these findings.

7.2 Theoretical implications: the roles are coupled in the technology but not in the research

The coupled dual-use framework (Section 2.3) proposed three couplings. The combined evidence supports each of them as a property of the technology, while showing that the research community has absorbed them unevenly.

Proposition C1 (every defensive deployment creates a target) is supported by the critical review threat-intelligence and SOC pipelines are retrieval systems over attacker-controllable text, the configuration that indirect prompt injection exploits (Greshake et al., 2023; Yi et al., 2025) and its neglect is the clearest bibliometric finding of this study. Defensive studies foreground attacks on the defending model in only 2.5–11.1% of cases, depending on the application area, and the co-citation analysis shows that defensive and attack research draw on different intellectual bases. This is the most consequential gap in the field, because the security value of a defensive LLM depends on its robustness to the content it is deployed to analyse.

Proposition C2 (offensive capability is re-purposed as defence) is supported for technical offence but not for human-targeted offence. Technical offence is not a separate research community, penetration testing is grouped into the code and vulnerability, and the primary offensive research is cited alongside defensive surveys and datasets. Human-targeted offence has a different behavior: while phishing, social engineering and deepfakes group together in the governance/ethics box, the generation of content is studied by distinct communities from the detection of content in the phishing box. The coupling at the level of individual results is well illustrated by CyberGym's discovery of 34 zero day vulnerabilities in the process of measuring offensive capability (Wang et al., 2025). Offence or defence, sometimes it depends on who runs it; again, a dual-use argument as articulated by Brundage et al. (2018).

The jailbreak methods literature has evolved from hand-crafted prompts to automated generators (Liu et al., 2023a; Zou et al., 2023; Chao et al., 2025), and from inferring and bypassing provider defences to fully automated systems (Deng et al., 2024). The target–weapon overlap, also quite small but not negligible, is the “bibliometric signature” (131 documents in all three roles), which is confined to automated red-teaming and misuse studies.

These results complement the offence–defence–balance perspective (Garfinkel and Dafoe, 2019). The balance is not just a function of what the technology can do, but also where the effort is going. The literature where LLM attacks and LLM defense are both written, evolves predictably: LLM defense is deployed and analyzed for attack detection, while LLM attack is written and studied before its attack surface is understood.

7.3 Why reported results diverge by an order of magnitude

A 90% offensive success rate is displayed in figure 9, ranging from 0% to 90%. Most of this spread can be attributed to four structural causes, none of which have to do exclusively with the ability of the models.

The first is speed of publication. Preprints make up 35.9% of the corpus and the norm is to publish an attack/defence within weeks after it is conceived. Such time is not available for adaptive-attack evaluation, repeated trials, or utility measurement, and abstract-level indicators of each of these practices are found in only 4% of target-role documents (Section 5.5). Consequences: Under principled evaluation, defences are less effective than reported, as shown by Jia et al. (2026). The second is related to the instability of the system under test. Commercial models are continually updated under stable API names, so a defence against one model is no guarantee against a different one a month later (Abdelnabi et al., 2026) as the staleness of time identified by their model. The practical implication of the 400 run study is that the exploitation rates are 25–85% in four different models on the same target, meaning that any 1 run result encompasses an unknown larger than many of the effects being reported. The third one is the lack of a common language of threats. There are different ways of operationalising the attack success rate in almost all studies and same metric labels are used for different judges, aggregation rules and threat models (Solanke, 2026). Yu et al. (2026) demonstrate that benchmarks that do not distinguish between reconnaissance and exploitation result in cascaded errors that inflate the assessment of autonomy, while deflating the assessment of exploitation capability. The fourth is incentives. Successful replications are not necessarily publishable, but a failed replication is. The fact that no evaluation-methodology study is cited in the most-cited documents of the field (Table 7) is attributable to this asymmetry, which is why the most careful of all the studies included in the evidence base are among the newest (Dai et al., 2025; Mezzi et al., 2025; Erdem, 2026; Jia et al., 2026).

Another, more venerable reason for the intrusion-detection literature has been documented for years: continued use of saturated datasets that have shortcomings well known for years (Tavallae et al., 2009; Sommer and Paxson, 2010). The presence of these datasets in the co-cited intellectual base indicates that evaluation issues of machine learning-based detection research are transferred. The presence of these datasets in the co-cited intellectual base suggests that evaluation issues of the machine learning based detection research are transferred.

7.4 Offence and defence are co-evolving, and the gap is not closing

When the milestones of Section 6 are set on a common time scale, there are two patterns. Defences have only addressed the attacks that already exist: prompt-level defences acted against indirect prompt injection; guard models acted against automated jailbreaking; and alignment-level defences have responded to the demonstration that prompt-level defences weren't sufficient. This is a normal sequence, it is not a criticism. The second pattern is concerning. The current generation of defence has failed to defeat them with some new attack capability, but rather with an attack against the structure of the defence itself. Evasion tuned for guard models (Hackett et al., 2025), evasion from the difference between the guardrail's inspection window and the model's context window (Zhou et al., 2026), and perturbations that the guardrail can't decode but the model can (Wang et al., 2026). These attacks made no advances in model capability. The same can be said of every layer of the canonical pipeline of defence: it has a published bypass in the same literature as the defence itself (see Figure 10 above). The good news in the reading is that none of the defence families reviewed here have proven to be immune to a determined adaptive attacker and most haven't been challenged by one.

7.5 Practical implications

There are three implications for organisations that deploy LLM-based systems. Firstly, the published attack success rates should be considered a characteristic of an experimental design and not a characteristic of a model. Practitioners should first determine if vulnerability context was reported and the number of trials conducted, and whether defences were present (Figure 9) before taking action on a reported figure. Second,

prompt injection is to be assumed to be unmitigated; systems should be designed to keep the blast radius to a minimum. The load-bearing controls are architectural in that they limit what can be done in a successful attack, but do not attempt to stop an attack from occurring. (Saltzer and Schroeder, 1975) has a history of bypass. This is especially true for deployments to the defensive side: If a SOC agent can perform response actions based on the content of the logs, it can be considered as an externally reachable component. Thirdly, the price of defending legitimate users should be calculated prior to deployment. This false-positive rate is substantial (Maiorano, 2026) and highlights the fact that the equivalent issue in secure code generation (techniques that reduced base model functionality by over 50% - Dai et al., 2025) routine practice has been ignored rather than neglected.

7.6 Implications for policy and standardisation

The gap between what regulation requires and what the technical literature can supply is measurable. Mapping benchmark content against the EU AI Act taxonomy reveals no coverage of the capabilities central to loss-of-control scenarios (Prandi et al., 2025). Because several governance frameworks, including the NIST AI Risk Management Framework (NIST, 2023), make measurement and testing central to risk management, and because benchmark results are self-reported and difficult to verify (Cheng et al., 2025), compliance claims currently rest on measurements that the field itself does not fully trust. Verifiable evaluation (Schnabl et al., 2025) addresses the verification half of this problem but not the coverage half. Our findings add one specific requirement: governance instruments for defensive AI should require evidence that the defending system has been tested against manipulation through the content it processes.

7.7 Research agenda

Table 16 lists eight research directions. Each is grounded in both the bibliometric indicators (Section 5) and the critical evidence (Section 6), and each is assessed for feasibility by research groups without frontier-scale compute.

Table 16. Research directions, supporting evidence and feasibility.

No.	Direction	Bibliometric evidence	Critical evidence	Feasibility without frontier compute
D1	Adaptive-adversary evaluation as standard practice	Adaptive attacks mentioned in 3.2% of target-role abstracts	Athalye et al., 2018; Jia et al., 2026; Tamuka et al., 2026	High
D2	Security of defensive LLM deployments (testing coupling C1)	2.5–11.1% of defensive studies address prompt injection or jailbreaks	Greshake et al., 2023; Mezzi et al., 2025; Yi et al., 2025	High
D3	Multilingual and low-resource guardrail robustness	3.3% of target-role abstracts mention multilingual evaluation; Central Asia < 1% of output	Solanke, 2026; Wang et al., 2026	High
D4	Longitudinal security measurement across model versions	1.3% of target-role abstracts mention longitudinal evaluation	Abdelnabi et al., 2026; Erdem, 2026	High
D5	Security–utility frontier characterisation	Utility or false positives mentioned in 13.9% of target-role abstracts	Dai et al., 2025; Li et al., 2025; Maiorano, 2026	High
D6	Cross-session and long-horizon threat benchmarks	Agentic terms are the fastest-growing keywords	Chu, 2026; Deng et al., 2026	Medium
D7	Human factors in AI-assisted security operations	Human-subject components in 2.0% of abstracts	Parasuraman and Riley, 1997; Jabir et al., 2025; Mezzi et al., 2025	High
D8	Attacker economics and verifiable evaluation	Weapon role is 8.3% of output and lacks its own intellectual base	Nakatani, 2025; Schnabl et al., 2025; Gonzaga et al., 2026; Happe et al., 2026	Medium to low

D1: Adaptive evaluation. Almost all defence evaluations test against a fixed set of attacks chosen before the defence was designed. Until adaptive evaluation is standard, published defence numbers should be treated as upper bounds, and tooling that makes adaptive evaluation inexpensive would be a concrete, high-value contribution (Athalye et al., 2018; Jia et al., 2026).

D2: Securing the shield. The most distinctive direction arising from this review is to evaluate defensive LLM systems threat-intelligence extractors, alert triage agents and detection-rule generators against injection and poisoning through the reports, logs and code they consume. The bibliometric evidence shows that this is rarely done, and the critical evidence shows why it matters: these systems are retrieval pipelines over attacker-controllable text (Greshake et al., 2023), and they are already known to be overconfident (Mezzi et al., 2025).

D3: Multilingual guardrail robustness. Guard models are trained predominantly on English adversarial corpora, whereas the models behind them are multilingual, and 165 of 195 safety benchmarks are English-only (Solanke, 2026). The representation asymmetry documented for character perturbations (Wang et al., 2026) plausibly extends to language: a guardrail may fail to recognise an attack that the protected model understands perfectly. We found no systematic study of this question for Central Asian, South Asian or Turkic languages. Translating established jailbreak and injection corpora (Shen et al., 2024; Yi et al., 2025) into Uzbek, Russian, Kazakh and Urdu and measuring refusal and detection degradation across open-weight models would be novel, inexpensive and directly publishable and it would address the geographic concentration identified in Section 5.1.

D4: Longitudinal measurement. Repeated measurement of a fixed attack corpus against fixed API endpoints over twelve months would produce the first longitudinal security dataset in this field and requires only API access and patience; it would quantify the temporal staleness that currently undermines comparison (Abdelnabi et al., 2026; Erdem, 2026).

D5: The security–utility frontier. Guardrail papers report detection rates but rarely what the guardrail does to legitimate users. Where measured, the results are unflattering (Dai et al., 2025; Li et al., 2025; Maiorano, 2026). Reporting the full security–utility frontier, rather than a single operating point, would greatly improve the decision relevance of the literature.

D6: Long-horizon threats. Memory poisoning that manifests three sessions later cannot be detected by a single-turn benchmark. No current benchmark covers cross-session cumulative or sub-session-stack failure modes (Chu, 2026), although such compound threats have been demonstrated in deployed agents (Deng et al., 2026). This is the largest structural gap in the evaluation ecosystem and the one most likely to matter as agents move into production.

D7: The analyst in the loop. LLMs are overconfident in threat intelligence (Mezzi et al., 2025), yet the effect of overconfident AI output on analyst trust calibration, automation bias and triage error rates is barely studied. Automation-bias research predicts misuse and disuse when trust is miscalibrated (Parasuraman and Riley, 1997), and human-factors methodology has already been applied successfully on the attack side (Jabir et al., 2025). The defensive counterpart of the de-skilling effect measured for attackers (Mishra and Varshney, 2025) what a defensive copilot does to novice analysts remains unmeasured. This direction requires human-subject studies rather than compute and is well suited to university settings.

D8: Attacker economics and verifiable evaluation. Autonomous penetration testing is reported at costs of roughly USD 0.30–0.60 per run (Nakatani, 2025) and at costs competitive with professional testers per exploited vulnerability (Happe et al., 2026), and the attacker's optimal strategy for social engineering has been formalised (Gonzaga et al., 2026), but no study models how a sustained reduction in per-target attack cost changes target selection at population scale, a question that security economics is well placed to address (Anderson and Moore, 2006). In parallel, attestable audits in trusted execution environments (Schnabl et al., 2025) respond to the concern that benchmark results are self-reported and unverifiable (Cheng et al., 2025); this direction is technically demanding and is listed for completeness.

7.8 Limitations

Several limitations qualify these findings. First, the bibliometric stage relies on a single database. OpenAlex was chosen for its coverage of preprints and its openness, but its reference lists and metadata are less

complete than those of curated databases (Alperin et al., 2024; Culbert et al., 2025); only 2,693 of the 7,369 documents had reference lists available for co-citation analysis, and five of the 100 most co-cited records had to be excluded because of corrupted metadata. A replication using Scopus or Web of Science would test whether the cluster structure is database-dependent. Second, the search string favours explicit security vocabulary: its recall against an independently compiled benchmark set was 79.3% for in-scope items, and agent-security and secure-code studies are under-represented in the maps. Third, the relevance screen has a precision of 84.0%, so roughly one record in six is peripheral; such records mostly fall below the occurrence thresholds of the maps, but they inflate the performance counts. Fourth, role classification and evaluation indicators are dictionary-based and operate on titles and abstracts; they measure what studies foreground, not what full texts contain, and the weapon dictionary has the lowest precision. The screening and role-precision checks were performed by a single coder, and the labelled samples are released so that they can be re-coded independently. Fifth, the corpus is English-language a limitation that this review shares with the evidence it criticises (direction D3) and the 2026 data are partial, so trends in the final year should be read as incomplete rather than as a slowdown. Finally, the critical review reports results as stated by the primary studies; Section 7.3 argues that many of these results are not directly comparable, and we have not reproduced any of them.

8. Conclusion

This article set out to establish whether research on large language models in cybersecurity reflects the dual-use character of the technology. By combining a bibliometric mapping of 7,369 documents with a critical review of the evidence, it shows that the literature has matured rapidly but unevenly. Taxonomies of attacks are abundant and largely convergent; defensive applications have progressed from proofs of concept to systems with credible reported performance; and offensive capability is real but constrained, bottlenecked at reconnaissance rather than exploitation (Yu et al., 2026), with headline figures whose qualifiers are often omitted (Fang et al., 2024a).

The central finding is that the three roles of LLMs target, weapon and shield are coupled in the technology but largely separate in the research. Fewer than one document in twelve addresses more than one role; defensive studies rarely examine attacks on the defending model; and attack and defence research rest on different intellectual bases. At the same time, measurement has not matured. Attack success rates for comparable tasks span an order of magnitude; defences are evaluated against attacks that predate them (Jia et al., 2026); the cost of defence to legitimate users is largely unreported (Dai et al., 2025; Maiorano, 2026); and benchmarks age faster than they are maintained, with most available only in English (Solanke, 2026).

These findings suggest that the most valuable contributions over the next several years may not be new attacks or new defences, of which there is a healthy supply, but better instruments for determining whether either works and, in particular, evaluations that treat every defensive LLM as the attack surface it is. The coupled dual-use framework, the released bibliometric data and the research agenda offered here are intended to support that shift.

Declarations

Funding: This research received no external funding.

Conflicts of interest: The authors declare no conflicts of interest.

Data availability: The OpenAlex query, harvesting and analysis scripts, cleaned corpus identifiers, screening sample with labels, thesaurus, role dictionaries, VOSviewer map and network files and the reference verification log are available as supplementary material and will be deposited in a public repository upon acceptance.

Appendix A. Screening rule and role dictionaries

The relevance screen retained a record when its title or abstract matched the technology pattern T and the security pattern S, and at least one of the two patterns matched the title. All patterns are case-insensitive regular expressions applied to titles and abstracts reconstructed from OpenAlex.

Table 17. Screening and role-classification patterns.

Pattern	Regular-expression terms (abridged)
T (technology)	large language model(s); LLM(s); ChatGPT; GPT-3/4/4o; generative AI or generative artificial intelligence; foundation model(s); language model(s); Claude; Llama; Gemini; BERT
S (security)	cyber security; cyber-attack(s); cyber threat(s); prompt injection(s); jailbreak*; phishing; malware; ransomware; penetration test*; intrusion detection; threat intelligence; vulnerabilit*; exploit*; adversarial attack(s)/prompt(s)/example(s); red-team*; backdoor*; data poisoning; poisoning attack(s); guardrail(s); security operation(s); incident response; social engineering; fraud; deepfake(s); privacy attack(s); membership inference; secure code; attack surface; threat model*; cyber defence; anomaly detection; spam
Target role	jailbreak*; prompt injection; prompt leak*; backdoor*; data/training/model/RAG/knowledge/memory poisoning; poisoned; guardrail(s); safety alignment; red-team*; adversarial prompt(s); membership inference; training-data extraction; model extraction or stealing; attack(s) on/against (large) language models or LLMs; LLM security/safety/vulnerabilit*; LLM or agent security/safety; harmful content/output
Weapon role	generation/crafting/creation/automation/writing of (convincing, realistic, targeted, personalised) phishing, spear-phishing, malware, ransomware, exploit(s), malicious code or polymorphic code; AI-/LLM-/GPT-generated, -crafted, -powered, -enabled, -driven or -assisted phishing, malware, social engineering, attacks, hacking or exploits; malicious use(s) or application(s) of LLMs, AI, generative AI or ChatGPT; weaponis*; offensive security/AI/cyber/use/capabilit*; penetration test*; pentest*; autonomous hacking/exploitation/attack*; exploit generation; LLM- or AI-based attacker(s), hacking or red-team*; WormGPT; FraudGPT; dual-use
Shield role	intrusion detection; threat intelligence; CTI; vulnerability detection/repair/discovery/assessment; malware detection/analysis/classification; phishing (email) detection; anomaly detection; log analysis/parsing; incident response; security operations; SOC; threat detection/hunting; secure code generation; program repair; spam detection; fraud detection; forensic*; cyber defence; detection rule(s)

Appendix B. Keyword thesaurus

Table 18. Thesaurus applied before keyword co-occurrence analysis.

Original keyword(s)	Merged label
jailbreak attacks; jailbreaking; LLM jailbreaking; jailbreak attack; jailbreaks	jailbreak
large language model security; LLM security	LLM security
large language model safety; LLM safety	LLM safety
threat intelligence; cyber threat intelligence (CTI)	cyber threat intelligence
intrusion detection systems; network intrusion detection; intrusion detection system	intrusion detection
phishing email detection	phishing detection
GPT-4o	GPT-4
AI agents; LLM-based agents; large language model agents	LLM agents
retrieval augmented generation; RAG	retrieval-augmented generation
adversarial attack	adversarial attacks
backdoor attack	backdoor attacks
hallucination	hallucinations
Llama 2; Llama 3	Llama
explainability	explainable AI
software vulnerability detection	vulnerability detection
false positives	false-positive reduction
prompt injection attacks	prompt injection
Removed generic terms: large language models; large language model; LLMs; LLM; artificial intelligence; generative AI; generative artificial intelligence; natural language processing; machine learning; deep learning	—

References

- Abdelnabi, S., Hicks, C., Rieck, K. and Sadeghi, A.-R. (2026) 'Measuring Security Without Fooling Ourselves: Why Benchmarking Agents Is Hard', *arXiv* [Preprint]. doi: 10.48550/arXiv.2605.22568.
- Alperin, J.P., Portenoy, J., Demes, K., Larivière, V. and Haustein, S. (2024) 'An analysis of the suitability of OpenAlex for bibliometric analyses', *arXiv* [Preprint]. doi: 10.48550/arXiv.2404.17663.
- Anderson, R. and Moore, T. (2006) 'The Economics of Information Security', *Science*, 314(5799), pp. 610–613. doi: 10.1126/science.1130992.
- Arp, D., Quiring, E., Pendlebury, F., Warnecke, A., Pierazzi, F., Wressnegger, C. et al. (2022) 'Dos and Don'ts of Machine Learning in Computer Security', in *Proc. 31st USENIX Security Symposium*. doi: 10.48550/arXiv.2010.09470.
- Athalye, A., Carlini, N. and Wagner, D. (2018) 'Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples', *arXiv* [Preprint]. doi: 10.48550/arXiv.1802.00420.
- Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P. et al. (2020) 'Language Models are Few-Shot Learners', *arXiv* [Preprint]. doi: 10.48550/arXiv.2005.14165.
- Brundage, M., Avin, S., Clark, J., Toner, H., Eckersley, P., Garfinkel, B. et al. (2018) 'The Malicious Use of Artificial Intelligence: Forecasting, Prevention, and Mitigation', *arXiv* [Preprint]. doi: 10.48550/arXiv.1802.07228.
- Callon, M., Courtial, J.-P., Turner, W.A. and Bauin, S. (1983) 'From translations to problematic networks: An introduction to co-word analysis', *Social Science Information*, 22(2), pp. 191–235. doi: 10.1177/053901883022002003.
- Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D. et al. (2019) 'On Evaluating Adversarial Robustness', *arXiv* [Preprint]. doi: 10.48550/arXiv.1902.06705.
- Chakraborty, S., Krishna, R., Ding, Y. and Ray, B. (2022) 'Deep Learning Based Vulnerability Detection: Are We There Yet?', *IEEE Transactions on Software Engineering*, 48(9), pp. 3280–3296. doi: 10.1109/TSE.2021.3087402.
- Chao, P., Robey, A., Dobriban, E., Hassani, H., Pappas, G.J. and Wong, E. (2025) 'Jailbreaking Black Box Large Language Models in Twenty Queries', in *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pp. 23–42. doi: 10.1109/SaTML64287.2025.00010.
- Chen, Y., Cui, M., Wang, D., Cao, Y., Yang, P., Jiang, B. et al. (2024) 'A survey of large language models for cyber threat detection', *Computers & Security*, 145, 104016. doi: 10.1016/j.cose.2024.104016.
- Chen, S., Zharmagambetov, A., Wagner, D. and Guo, C. (2025a) 'Meta-SecAlign: Training LLMs against Prompt Injection for Robust Agents', *arXiv* [Preprint]. doi: 10.48550/arXiv.2507.02735.
- Chen, S., Zharmagambetov, A., Mahloujifar, S., Chaudhuri, K., Wagner, D. and Guo, C. (2025b) 'SecAlign: Defending Against Prompt Injection with Preference Optimization', in *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2833–2847. doi: 10.1145/3719027.3744836.
- Cheng, Z., Wahnig, S., Gupta, R., Alam, S., Abdullahi, T., Ribeiro, J.A. et al. (2025) 'Benchmarking is Broken -- Don't Let AI be its Own Judge', *arXiv* [Preprint]. doi: 10.48550/arXiv.2510.07575.
- Chennabasappa, S., Nikolaidis, C., Song, D., Molnar, D., Ding, S., Wan, S. et al. (2025) 'LlamaFirewall: An open source guardrail system for building secure AI agents', *arXiv* [Preprint]. doi: 10.48550/arXiv.2505.03574.
- Chhabra, A., Datta, S., Nahin, S.K. and Mohapatra, P. (2026) 'Agentic AI Security: Threats, Defenses, Evaluation, and Open Challenges', *IEEE Access*, 14, pp. 49455–49482. doi: 10.1109/ACCESS.2026.3675554.
- Chu, K. (2026) 'A Systematic Survey of Security Threats and Defenses in LLM-Based AI Agents: A Layered Attack Surface Framework', *arXiv* [Preprint]. doi: 10.48550/arXiv.2604.23338.
- Correia, P.H.B., Achjjan, R.W., de Oliveira, D.E.G.C., Maria, Y.A., Hayashi, V.T., Lopes, M. et al. (2026) 'A Systematic Literature Review on LLM Defenses Against Prompt Injection and Jailbreaking: Expanding NIST Taxonomy', *arXiv* [Preprint]. doi: 10.48550/arXiv.2601.22240.
- Culbert, J.H., Hobert, A., Jahn, N., Haupka, N., Schmidt, M., Donner, P. et al. (2025) 'Reference coverage analysis of OpenAlex compared to Web of Science and Scopus', *Scientometrics*, 130(4), pp. 2475–2492. doi: 10.1007/s11192-025-05293-3.
- Muhammad Ramzan Ali, Dr Muhammad Bilal Qureshi, Dr Adnan Manzoor (2025) "CYBER SECURITY AND ARTIFICIAL INTELLIGENCE: REVOLUTIONARY TRENDS CREATING THE FUTURE". Zenodo. Available at: <https://doi.org/10.5281/zenodo.14911281>
- Dai, S.-C., Xu, J. and Tao, G. (2025) 'Rethinking the Evaluation of Secure Code Generation', *arXiv* [Preprint]. doi: 10.48550/arXiv.2503.15554.

- Damacena Duarte, J., Cândido, G.D., De Britto Filho, J.R.A., Souza Neto, J., Da Costa, E.J., Da Costa, J.P.J. et al. (2026) 'A Systematic Review of Prompt Injection Attacks on Large Language Models: Trends, Taxonomy, Evaluation, Defenses, and Opportunities', *IEEE Access*, 14, pp. 12875–12899. doi: 10.1109/ACCESS.2026.3656849.
- Das, B.C., Amini, M.H. and Wu, Y. (2025) 'Security and Privacy Challenges of Large Language Models: A Survey', *ACM Computing Surveys*, 57(6), pp. 1–39. doi: 10.1145/3712001.
- Deng, G., Liu, Y., Li, Y., Wang, K., Zhang, Y., Li, Z. et al. (2024) 'MASTERKEY: Automated Jailbreaking of Large Language Model Chatbots', in *Proceedings 2024 Network and Distributed System Security Symposium*. doi: 10.14722/ndss.2024.24188.
- Deng, X., Zhang, Y., Wu, J., Bai, J., Yi, S., Zou, Z. et al. (2026) 'Taming OpenClaw: Security Analysis and Mitigation of Autonomous LLM Agent Threats', *arXiv [Preprint]*. doi: 10.48550/arXiv.2603.11619.
- Ding, Y., Fu, Y., Ibrahim, O., Sitawarin, C., Chen, X., Alomair, B. et al. (2025) 'Vulnerability Detection with Code Language Models: How Far are We?', in *2025 IEEE/ACM 47th International Conference on Software Engineering (ICSE)*, pp. 1729–1741. doi: 10.1109/ICSE55347.2025.00038.
- Donthu, N., Kumar, S., Mukherjee, D., Pandey, N. and Lim, W.M. (2021) 'How to conduct a bibliometric analysis: An overview and guidelines', *Journal of Business Research*, 133, pp. 285–296. doi: 10.1016/j.jbusres.2021.04.070.
- Erdem, G.T. (2026) 'How Reliable Are AI Attackers Against a Fixed Vulnerable Target? A 400-Run Empirical Study of LLM Penetration Testing Consistency', *arXiv [Preprint]*. doi: 10.48550/arXiv.2605.30096.
- Eze, C.S. and Shamir, L. (2024) 'Analysis and Prevention of AI-Based Phishing Email Attacks', *Electronics*, 13(10), 1839. doi: 10.3390/electronics13101839.
- Fang, R., Bindu, R., Gupta, A. and Kang, D. (2024a) 'LLM Agents can Autonomously Exploit One-day Vulnerabilities', *arXiv [Preprint]*. doi: 10.48550/arXiv.2404.08144.
- Fang, R., Bindu, R., Gupta, A., Zhan, Q. and Kang, D. (2024b) 'LLM Agents can Autonomously Hack Websites', *arXiv [Preprint]*. doi: 10.48550/arXiv.2402.06664.
- Ferrag, M.A., Ndhlovu, M., Tihanyi, N., Cordeiro, L.C., Debbah, M., Lestable, T. et al. (2024) 'Revolutionizing Cyber Threat Detection With Large Language Models: A Privacy-Preserving BERT-Based Lightweight Model for IoT/IIoT Devices', *IEEE Access*, 12, pp. 23733–23750. doi: 10.1109/ACCESS.2024.3363469.
- Ferrag, M.A., Alwahedi, F., Battah, A., Cherif, B., Mechri, A., Tihanyi, N. et al. (2025) 'Generative AI in cybersecurity: A comprehensive review of LLM applications and vulnerabilities', *Internet of Things and Cyber-Physical Systems*, 5, pp. 1–46. doi: 10.1016/j.iotcps.2025.01.001.
- Ferrag, M.A., Tihanyi, N., Hamouda, D., Maglaras, L., Lakas, A. and Debbah, M. (2026) 'From prompt injections to protocol exploits: Threats in LLM-powered AI agents workflows', *ICT Express*, 12(2), pp. 353–383. doi: 10.1016/j.icte.2025.12.001.
- Ferrara, E. (2024) 'GenAI against humanity: nefarious applications of generative artificial intelligence and large language models', *Journal of Computational Social Science*, 7(1), pp. 549–569. doi: 10.1007/s42001-024-00250-1.
- Garfinkel, B. and Dafoe, A. (2019) 'How does the offense-defense balance scale?', *Journal of Strategic Studies*, 42(6), pp. 736–763. doi: 10.1080/01402390.2019.1631810.
- Gonzaga, K., Serra, S., Gomes, M. and Malta, S. (2026) 'AI-Powered Social Engineering: Emerging Attack Vectors, Vulnerabilities, and Multi-Layered Defense Strategies', *Computers*, 15(2), 128. doi: 10.3390/computers15020128.
- Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T. and Fritz, M. (2023) 'Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection', in *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*, pp. 79–90. doi: 10.1145/3605764.3623985.
- Gulyamov, S., Gulyamov, S., Rodionov, A., Khursanov, R., Mekhmonov, K., Babaev, D. et al. (2026) 'Prompt Injection Attacks in Large Language Models and AI Agent Systems: A Comprehensive Review of Vulnerabilities, Attack Vectors, and Defense Mechanisms', *Information*, 17(1), 54. doi: 10.3390/info17010054.
- Gupta, M., Akiri, C., Aryal, K., Parker, E. and Praharaj, L. (2023) 'From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy', *IEEE Access*, 11, pp. 80218–80245. doi: 10.1109/ACCESS.2023.3300381.
- Hackett, W., Birch, L., Trawicki, S., Suri, N. and Garraghan, P. (2025) 'Bypassing LLM Guardrails: An Empirical Analysis of Evasion Attacks against Prompt Injection and Jailbreak Detection Systems', *arXiv [Preprint]*. doi: 10.48550/arXiv.2504.11168.
- Happe, A. and Cito, J. (2023) 'Getting pwn'd by AI: Penetration Testing with Large Language Models', in *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 2082–2086. doi: 10.1145/3611643.3613083.

- Alimova Zebo1, Muhammad Ramzan Ali(PhD)2, Bakhronkulova Laylo3, Farzona Mirsodikova4, 5Irkinov Rasuljon (2025) "THE ROLE OF AI IN DEVELOPING GLOBAL CYBERSECURITY POLICIES". Zenodo. Available at: <https://doi.org/10.5281/zenodo.14899972>
- Happe, A., Kaplan, A. and Cito, J. (2026) 'LLMs as Hackers: Autonomous Linux Privilege Escalation Attacks', *Empirical Software Engineering*, 31(3). doi: 10.1007/s10664-025-10758-3.
- Hassanin, M., Keshk, M., Salim, S., Alsubaie, M. and Sharma, D. (2024) 'PLLM-CS: Pre-trained Large Language Model (LLM) for Cyber Threat Detection in Satellite Networks', *arXiv [Preprint]*. doi: 10.48550/arXiv.2405.05469.
- He, F., Zhu, T., Ye, D., Liu, B., Zhou, W. and Yu, P.S. (2025) 'The Emerged Security and Privacy of LLM Agent: A Survey with Case Studies', *ACM Computing Surveys*, 58(6), pp. 1–36. doi: 10.1145/3773080.
- Heiding, F., Schneier, B., Vishwanath, A., Bernstein, J. and Park, P.S. (2024) 'Devising and Detecting Phishing Emails Using Large Language Models', *IEEE Access*, 12, pp. 42131–42146. doi: 10.1109/ACCESS.2024.3375882.
- Hmimou, Y., Tabaa, M., Khiat, A. and Hidila, Z. (2025) 'A Multi-Agent System for Cybersecurity Threat Detection and Correlation Using Large Language Models', *IEEE Access*, 13, pp. 150199–150215. doi: 10.1109/ACCESS.2025.3602681.
- Shah, A.S.A., Ali, M.R., Vokhidova, M., Xusanov, C. and Musaeva, R., 2025. Charting the Future: Analyzing Innovative Strategies in Strategic Management for Sustainable Performance Excellence and Business Leadership in the Telecom Industry. *Business Ethics and Leadership*, 9(4), pp.195-208. [https://doi.org/10.61093/bel.9\(4\).195-208.2025](https://doi.org/10.61093/bel.9(4).195-208.2025)
- Ali, M., Peiseniece, L. and Mikelson, E., 2025, June. Through the green work commitment, a literature review examines the relationship between green HRM and environmental sustainability. In *Environment. Technology. Resources. Proceedings of the International Scientific and Practical Conference (Vol. 1, pp. 31-43)*. <https://doi.org/10.17770/etr2025vol1.8635>
- Ali, M., Peiseniece, L. and Mikelson, E., 2025, June. The mediating role of green work commitment in the relationship between green HR practices, AI elements, and environmental sustainability: a quantitative analysis. In *ENVIRONMENT. TECHNOLOGY. RESOURCES. Proceedings of the International Scientific and Practical Conference (Vol. 2, pp. 15-22)*. <https://doi.org/10.17770/etr2025vol2.8578>
- Hu, Y., Zou, F., Han, J., Sun, X. and Wang, Y. (2024) 'LLM-TIKG: Threat intelligence knowledge graph construction utilizing large language model', *Computers & Security*, 145, 103999. doi: 10.1016/j.cose.2024.103999.
- Jabir, R., Le, J. and Nguyen, C. (2025) 'Phishing Attacks in the Age of Generative Artificial Intelligence: A Systematic Review of Human Factors', *AI*, 6(8), 174. doi: 10.3390/ai6080174.
- Jia, Y., Shao, Z., Liu, Y., Jia, J., Song, D. and Gong, N. (2026) 'A Critical Evaluation of Defenses against Prompt Injection Attacks: [Dataset/Tool Paper]', in *Proceedings of the 31st ACM Symposium on Access Control Models and Technologies*, pp. 265–270. doi: 10.1145/3750555.3811884.
- Li, H., Guo, D., Fan, W., Xu, M., Huang, J., Meng, F. et al. (2023) 'Multi-step Jailbreaking Privacy Attacks on ChatGPT', in *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 4138–4153. doi: 10.18653/v1/2023.findings-emnlp.272.
- Li, Z., Dutta, S. and Naik, M. (2024) 'IRIS: LLM-Assisted Static Analysis for Detecting Security Vulnerabilities', *arXiv [Preprint]*. doi: 10.48550/arXiv.2405.17238.
- Li, H., Liu, X., Zhang, N. and Xiao, C. (2025) 'PIGuard: Prompt Injection Guardrail via Mitigating Overdefense for Free', in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 30420–30437. doi: 10.18653/v1/2025.acl-long.1468.
- Li, H., Yang, Y., Suh, G.E., Zhang, N. and Xiao, C. (2026) 'ReasAlign: Reasoning Enhanced Safety Alignment against Prompt Injection Attack', *arXiv [Preprint]*. doi: 10.48550/arXiv.2601.10173.
- Liu, X., Xu, N., Chen, M. and Xiao, C. (2023a) 'AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models', *arXiv [Preprint]*. doi: 10.48550/arXiv.2310.04451.
- Liu, Y., Jia, Y., Geng, R., Jia, J. and Gong, N.Z. (2023b) 'Formalizing and Benchmarking Prompt Injection Attacks and Defenses', *arXiv [Preprint]*. doi: 10.48550/arXiv.2310.12815.
- Liu, X., Liang, J., Yan, Q., Jang, J., Mao, S., Ye, M. et al. (2025a) 'CyLens: Towards Reinventing Cyber Threat Intelligence in the Paradigm of Agentic Large Language Models', *arXiv [Preprint]*. doi: 10.48550/arXiv.2502.20791.
- Liu, Z., Huang, L., Zhang, J., Liu, D., Tian, Y. and Shao, J. (2025b) 'PACEbench: A Framework for Evaluating Practical AI Cyber-Exploitation Capabilities', *arXiv [Preprint]*. doi: 10.48550/arXiv.2510.11688.
- Khan, S., Ali, M., Akram, M., Azizova, N., Vokhidova, M. and Isamukhamedova, G., 2025, June. Exploring the impact of digital transformation on business model innovation in the retail sector. In *Environment. Technology. Resources*.

Proceedings of the International Scientific and Practical Conference (Vol. 1, pp. 281-288).
<https://doi.org/10.17770/etr2025vol1.8640>

Muhammad Ramzan ALI, Muhammad Sharjeel Aziz, Muhammad Younis (2026) "FOSTERING ENVIRONMENTAL SUSTAINABILITY THROUGH GREEN HUMAN RESOURCE MANAGEMENT, GREEN LEADERSHIP, AND WORK ENGAGEMENT: EXAMINING THE EMERGING ROLE OF ARTIFICIAL INTELLIGENCE". Zenodo. Available at: <https://doi.org/10.5281/zenodo.22897287>

Ali, M.R. and Iqbal, S.M.J., 2022. Examining the mediating influence of green behavior on the impact of green HR practices on environmental sustainability. *iRASD Journal of Management*, 4(3), pp.479-494.

Luo, J., Dai, J., Chen, Z., Xu, J., Wang, W., Duan, Y. et al. (2026) 'The Emergence of Autonomous Penetration Capabilities in Large Language Model-Powered AI Systems', *arXiv* [Preprint]. doi: 10.48550/arXiv.2606.13079.

Maiorano, A.C. (2026) 'Evaluating Prompt Injection Defenses for Educational LLM Tutors: Security-Usability-Latency Trade-offs', *arXiv* [Preprint]. doi: 10.48550/arXiv.2605.06669.

Mezzi, E., Massacci, F. and Tuma, K. (2025) 'Large Language Models Are Unreliable for Cyber Threat Intelligence', in *Lecture Notes in Computer Science*, pp. 343–364. doi: 10.1007/978-3-032-00627-1_17.

Mishra, R. and Varshney, G. (2025) 'Jailbreaking Generative AI: Multivector Phishing Threats and Transformer based Defenses', *arXiv* [Preprint]. doi: 10.48550/arXiv.2507.12185.

Khan, S., Ali, M., Akram, M., Azizova, N., Vokhidova, M. and Isamukhamedova, G., 2025, June. Exploring the impact of digital transformation on business model innovation in the retail sector. In *Environment. Technology. Resources. Proceedings of the International Scientific and Practical Conference* (Vol. 1, pp. 281-288). <https://doi.org/10.17770/etr2025vol1.8640>

Muhammad Ramzan ALI, Muhammad Sharjeel Aziz, Muhammad Younis (2026) "FOSTERING ENVIRONMENTAL SUSTAINABILITY THROUGH GREEN HUMAN RESOURCE MANAGEMENT, GREEN LEADERSHIP, AND WORK ENGAGEMENT: EXAMINING THE EMERGING ROLE OF ARTIFICIAL INTELLIGENCE". Zenodo. Available at: <https://doi.org/10.5281/zenodo.22897287>

Ali, M.R. and Iqbal, S.M.J., 2022. Examining the mediating influence of green behavior on the impact of green HR practices on environmental sustainability. *iRASD Journal of Management*, 4(3), pp.479-494.

Moustafa, N. and Slay, J. (2015) 'UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)', in *2015 Military Communications and Information Systems Conference (MilCIS)*, pp. 1–6. doi: 10.1109/MilCIS.2015.7348942.

Nakatani, S. (2025) 'RapidPen: Fully Automated IP-to-Shell Penetration Testing with LLM-based Agents', *arXiv* [Preprint]. doi: 10.48550/arXiv.2502.16730.

Narajala, V.S. and Narayan, O. (2025) 'Securing Agentic AI: A Comprehensive Threat Model and Mitigation Framework for Generative AI Agents', in *2025 8th International Conference on Algorithms, Computing and Artificial Intelligence (ACAI)*, pp. 1–6. doi: 10.1109/ACAI68217.2025.11406310.

National Institute of Standards and Technology (NIST) (2023) *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. National Institute of Standards and Technology, NIST AI 100-1. Gaithersburg, MD: National Institute of Standards and Technology. doi: 10.6028/NIST.AI.100-1.

Page, M.J., McKenzie, J.E., Bossuyt, P.M., Boutron, I., Hoffmann, T.C., Mulrow, C.D. et al. (2021) 'The PRISMA 2020 statement: an updated guideline for reporting systematic reviews', *BMJ*, n71. doi: 10.1136/bmj.n71.

Pai, A. (2026) 'Evaluating Prompting-Based Defenses Against Domain-Camouflaged Injection Attacks', *arXiv* [Preprint]. doi: 10.48550/arXiv.2606.18530.

Parasuraman, R. and Riley, V. (1997) 'Humans and Automation: Use, Misuse, Disuse, Abuse', *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 39(2), pp. 230–253. doi: 10.1518/001872097778543886.

Pearce, H., Ahmad, B., Tan, B., Dolan-Gavitt, B. and Karri, R. (2022) 'Asleep at the Keyboard? Assessing the Security of GitHub Copilot's Code Contributions', in *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 754–768. doi: 10.1109/SP46214.2022.9833571.

Pearce, H., Tan, B., Ahmad, B., Karri, R. and Dolan-Gavitt, B. (2023) 'Examining Zero-Shot Vulnerability Repair with Large Language Models', in *2023 IEEE Symposium on Security and Privacy (SP)*, pp. 2339–2356. doi: 10.1109/SP46215.2023.10179324.

- Peng, J., Cui, L., Huang, K., Yang, J. and Ray, B. (2025) 'CWEval: Outcome-driven Evaluation on Functionality and Security of LLM Code Generation', in *2025 IEEE/ACM International Workshop on Large Language Models for Code (LLM4Code)*, pp. 33–40. doi: 10.1109/LLM4Code66737.2025.00009.
- Perez, F. and Ribeiro, I. (2022) 'Ignore Previous Prompt: Attack Techniques For Language Models', *arXiv [Preprint]*. doi: 10.48550/arXiv.2211.09527.
- Tehmina Rafi, Muhammad Ramzan Ali, Zulaykho Kadirova, Usama Yousaf (2025) "IMPACT OF BOARD GENDER DIVERSITY, FOREIGN OWNERSHIP, AND OWNERSHIP CONCENTRATION ON CORPORATE FINANCIAL DECISIONS AND FIRM VALUE: EVIDENCE FROM THE TEXTILE SECTOR OF PAKISTAN". Zenodo. Available at: <https://doi.org/10.5281/zenodo.17421976>
- Ali, M., Kadirova, Z. and Ismoyilov, B., 2024, June. Practicing green HRM to achieve environmental sustainability. In *Environment. Technology. Resources. Proceedings of the International Scientific and Practical Conference* (Vol. 1, pp. 24-33). <https://doi.org/10.17770/etr2024vol1.7961>
- MUHAMMAD, R. A., BAKHTIYOR, I., MAHLIYO, M. & MUHAMMAD, F. A. 2023. Role of Servant Leadership in Work Performance: Identifying the Mediating Relation with Employee Ambidexterity and Moderating with Competitive Orientation. *Oeconomia Copernicana*, 14, 181-211.
- Prandi, M., Suriani, V., Pierucci, F., Galisai, M., Nardi, D. and Bisconti, P. (2025) 'Bench-2-CoP: Can We Trust Benchmarking for EU AI Compliance?', *arXiv [Preprint]*. doi: 10.48550/arXiv.2508.05464.
- Pribisalić, M. and Martinčić-Ipšić, S. (2026) 'Security and Privacy of Large Language Models: Threat Taxonomy, Ethical Implications, and Governance', *AI*, 7(5), 152. doi: 10.3390/ai7050152.
- Priem, J., Piwowar, H. and Orr, R. (2022) 'OpenAlex: A fully-open index of scholarly works, authors, venues, institutions, and concepts', *arXiv [Preprint]*. doi: 10.48550/arXiv.2205.01833.
- Saltzer, J. and Schroeder, M. (1975) 'The protection of information in computer systems', *Proceedings of the IEEE*, 63(9), pp. 1278–1308. doi: 10.1109/PROC.1975.9939.
- Schmitt, M. and Flechais, I. (2024) 'Digital deception: generative artificial intelligence in social engineering and phishing', *Artificial Intelligence Review*, 57(12). doi: 10.1007/s10462-024-10973-2.
- Schnabl, C., Hugenroth, D., Marino, B. and Beresford, A.R. (2025) 'Attestable Audits: Verifiable AI Safety Benchmarks Using Trusted Execution Environments', *arXiv [Preprint]*. doi: 10.48550/arXiv.2506.23706.
- Sharafaldin, I., Habibi Lashkari, A. and Ghorbani, A.A. (2018) 'Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization', in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, pp. 108–116. doi: 10.5220/0006639801080116.
- Ali, M.R., Peiseniece, L., Mikelson, E., Shah, A.S.A. and Kadirova, Z., 2026. Co-Creating Sustainable Innovation From Within: Green HRM Practices Enabling Employee Green Behavior in KIBS Ecosystems Across Pakistan, China, And Uzbekistan. *Journal of Management Practices, Humanities and Social Sciences*, 10(2), pp.263-277. <https://doi.org/10.33152/jmphss-10.2.7>
- Shen, X., Chen, Z., Backes, M., Shen, Y. and Zhang, Y. (2024) 'Do Anything Now': Characterizing and Evaluating In-The-Wild Jailbreak Prompts on Large Language Models', in *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pp. 1671–1685. doi: 10.1145/3658644.3670388.
- Shi, T., Zhu, K., Wang, Z., Jia, Y., Cai, W., Liang, W. et al. (2025) 'PromptArmor: Simple yet Effective Prompt Injection Defenses', *arXiv [Preprint]*. doi: 10.48550/arXiv.2507.15219.
- Small, H. (1973) 'Co-citation in the scientific literature: A new measure of the relationship between two documents', *Journal of the American Society for Information Science*, 24(4), pp. 265–269. doi: 10.1002/asi.4630240406.
- Solanke, A.A. (2026) 'AISafetyBenchExplorer: A Metric-Aware Catalogue of AI Safety Benchmarks Reveals Fragmented Measurement and Weak Benchmark Governance', *arXiv [Preprint]*. doi: 10.48550/arXiv.2604.12875.
- Sommer, R. and Paxson, V. (2010) 'Outside the Closed World: On Using Machine Learning for Network Intrusion Detection', in *2010 IEEE Symposium on Security and Privacy*, pp. 305–316. doi: 10.1109/SP.2010.25.
- Su, H., Luo, J., Liu, C., Yang, X., Zhang, Y., Dong, Y. et al. (2025) 'A Survey on Autonomy-Induced Security Risks in Large Model-Based Agents', *arXiv [Preprint]*. doi: 10.48550/arXiv.2506.23844.
- Tamuka, N., Mathonsi, T.E., Otieno, O.T., Maswikaneng, S., Muchenje, T. and Tshilongamulenzhe, T.M. (2026) 'Securing LLM-based agents against cyberattacks: a comprehensive survey on attack techniques and defense strategies', *Journal of Computer Virology and Hacking Techniques*, 22(1). doi: 10.1007/s11416-026-00622-3.

- Tavallaee, M., Bagheri, E., Lu, W. and Ghorbani, A.A. (2009) 'A detailed analysis of the KDD CUP 99 data set', in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 1–6. doi: 10.1109/CISDA.2009.5356528.
- AKRAM, M. F., ALI, M. R. & KADIROVA, Z. 2023a. ROLE OF CONSUMER DIGITAL LITERACY IN CONSUMER BEHAVIORAL INTENTIONS: MEDIATING EFFECT OF PERCEIVED EASE OF USE, USEFULNESS AND CUSTOMER ENGAGEMENT.
- AKRAM, M. F., ALI, M. R. & KADIROVA, Z. 2023b. Role of consumer digital literacy in consumer behavioral intentions: Mediating effect of perceived ease of use, usefulness and customer engagement. *Multidisciplinary Journal of Educational Research*, 13, 104-130.
- ALI, M. R., AKRAM, M. F., URKINBAYEV, T. A., TASHTAYEVA, S. K. & ARIPDJANOV, G. T. 2023. The Pathways to Achieve Green Economy and The Process of Sustainable Development Goals. *SGS-Engineering & Sciences*, 2.
- Muhammad Ramzan Ali, Dr. Liga Peiseniece, Dr. Elina Mikelsone (2025) "ARTIFICIAL INTELLIGENCE IN GREEN HUMAN RESOURCE MANAGEMENT: UNDERSTANDING THE MEDIATING ROLE OF EMPLOYEE ENGAGEMENT IN ACHIEVING ENVIRONMENTAL SUSTAINABILITY". Zenodo. Available at: <https://doi.org/10.5281/zenodo.15193336>.
- Traag, V.A., Waltman, L. and van Eck, N.J. (2019) 'From Louvain to Leiden: guaranteeing well-connected communities', *Scientific Reports*, 9(1). doi: 10.1038/s41598-019-41695-z.
- Ullah, S., Han, M., Pujar, S., Pearce, H., Coskun, A. and Stringhini, G. (2024) 'LLMs Cannot Reliably Identify and Reason About Security Vulnerabilities (Yet?): A Comprehensive Evaluation, Framework, and Benchmarks', in *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 862–880. doi: 10.1109/SP54263.2024.00210.
- van Eck, N.J. and Waltman, L. (2009) 'How to normalize cooccurrence data? An analysis of some well-known similarity measures', *Journal of the American Society for Information Science and Technology*, 60(8), pp. 1635–1651. doi: 10.1002/asi.21075.
- van Eck, N.J. and Waltman, L. (2010) 'Software survey: VOSviewer, a computer program for bibliometric mapping', *Scientometrics*, 84(2), pp. 523–538. doi: 10.1007/s11192-009-0146-3.
- Vassilev, A., Oprea, A., Fordyce, A. and Anderson, H. (2024) *Adversarial machine learning: A taxonomy and terminology of attacks and mitigations*. National Institute of Standards and Technology, NIST AI 100-2e2023. Gaithersburg, MD: National Institute of Standards and Technology. doi: 10.6028/NIST.AI.100-2e2023.
- Waltman, L., van Eck, N.J. and Noyons, E.C. (2010) 'A unified approach to mapping and clustering of bibliometric networks', *Journal of Informetrics*, 4(4), pp. 629–635. doi: 10.1016/j.joi.2010.07.002.
- Wang, Z., Shi, T., He, J., Cai, M., Zhang, J. and Song, D. (2025) 'CyberGym: Evaluating AI Agents' Real-World Cybersecurity Capabilities at Scale', *arXiv [Preprint]*. doi: 10.48550/arXiv.2506.02548.
- Wang, J., Zhu, Z. and Liu, C. (2026) 'The Asymmetric Vulnerability: Bypassing LLM Defenses via Guardrail-Model Mismatch', in *Proceedings of the ACM Web Conference 2026*, pp. 3078–3089. doi: 10.1145/3774904.3792438.
- Wei, A., Haghtalab, N. and Steinhardt, J. (2023) 'Jailbroken: How Does LLM Safety Training Fail?', in *Advances in Neural Information Processing Systems 36*, pp. 80079–80110. doi: 10.52202/075280-3508.
- Muhammad Ramzan Ali, Dr. Liga Peiseniece, Dr. Elina Mikelsone (2025) "ARTIFICIAL INTELLIGENCE IN GREEN HUMAN RESOURCE MANAGEMENT: UNDERSTANDING THE MEDIATING ROLE OF EMPLOYEE ENGAGEMENT IN ACHIEVING ENVIRONMENTAL SUSTAINABILITY". Zenodo. Available at: <https://doi.org/10.5281/zenodo.15193336>.
- Xie, Y., Yi, J., Shao, J., Curl, J., Lyu, L., Chen, Q. et al. (2023) 'Defending ChatGPT against jailbreak attack via self-reminders', *Nature Machine Intelligence*, 5(12), pp. 1486–1496. doi: 10.1038/s42256-023-00765-8.
- Xu, H., Wang, S., Li, N., Wang, K., Zhao, Y., Chen, K. et al. (2025) 'Large Language Models for Cyber Security: A Systematic Literature Review', *ACM Transactions on Software Engineering and Methodology*. doi: 10.1145/3769676.
- Yang, R., Cheng, M., Deng, G., Zhang, T., Wang, J. and Xie, X. (2025) 'PentestEval: Benchmarking LLM-based Penetration Testing with Modular and Stage-Level Design', *arXiv [Preprint]*. doi: 10.48550/arXiv.2512.14233.
- Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z. and Zhang, Y. (2024) 'A survey on large language model (LLM) security and privacy: The Good, The Bad, and The Ugly', *High-Confidence Computing*, 4(2), 100211. doi: 10.1016/j.hcc.2024.100211.
- Yeo, A. and Choi, D. (2025) 'Multimodal Prompt Injection Attacks: Risks and Defenses for Modern LLMs', *arXiv [Preprint]*. doi: 10.48550/arXiv.2509.05883.

- Yi, S., Liu, Y., Sun, Z., Cong, T., He, X., Song, J. et al. (2024) 'Jailbreak Attacks and Defenses Against Large Language Models: A Survey', *arXiv* [Preprint]. doi: 10.48550/arXiv.2407.04295.
- Ali, M.R., Peiseniece, L. & Mikelsone, E. Fostering environmental sustainability through green HRM green leadership and work commitment. *Discov Sustain* (2026). <https://doi.org/10.1007/s43621-026-04468-z>
- Yi, J., Xie, Y., Zhu, B., Kiciman, E., Sun, G., Xie, X. et al. (2025) 'Benchmarking and Defending against Indirect Prompt Injection Attacks on Large Language Models', in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, pp. 1809–1820. doi: 10.1145/3690624.3709179.
- Yu, L., Li, S., Zhou, M., Chu, G. and Guo, Y. (2026) 'Decoupling Reconnaissance and Exploitation: Measuring the Capability Boundaries of LLM-Based Web Penetration Testing', in *Lecture Notes in Computer Science*, pp. 607–618. doi: 10.1007/978-981-92-3423-3_49.
- Zhan, Q., Liang, Z., Ying, Z. and Kang, D. (2024) 'InjecAgent: Benchmarking Indirect Prompt Injections in Tool-Integrated Large Language Model Agents', in *Findings of the Association for Computational Linguistics ACL 2024*, pp. 10471–10506. doi: 10.18653/v1/2024.findings-acl.624.
- Bakhronkulova, L., Ali, M., Khabirova, Z., Azimjon, A., Zebo, A. and Muslima, A., 2025, June. Artificial Intelligence in Cybersecurity: Threats, Defenses, and Future Directions. In ENVIRONMENT. TECHNOLOGY. RESOURCES. Proceedings of the International Scientific and Practical Conference (Vol. 2, pp. 23-30). <https://doi.org/10.17770/etr2025vol2.8592>
- Zhang, H., Huang, J., Mei, K., Yao, Y., Wang, Z., Zhan, C. et al. (2024a) 'Agent Security Bench (ASB): Formalizing and Benchmarking Attacks and Defenses in LLM-based Agents', *arXiv* [Preprint]. doi: 10.48550/arXiv.2410.02644.
- Zhang, B., Tan, Y., Shen, Y., Salem, A., Backes, M., Zannettou, S. et al. (2024b) 'Breaking Agents: Compromising Autonomous LLM Agents Through Malfunction Amplification', *arXiv* [Preprint]. doi: 10.48550/arXiv.2407.20859.
- Zhang, J., Bu, H., Wen, H., Liu, Y., Fei, H., Xi, R. et al. (2025) 'When LLMs meet cybersecurity: a systematic literature review', *Cybersecurity*, 8(1). doi: 10.1186/s42400-025-00361-w.
- Zhou, Y., Zhu, C., Wang, J., He, X., Zhai, Y., Sun, K. et al. (2026) 'Prompt Overflow: What the Guardrail Inspects Is Not What the Model Infers', *arXiv* [Preprint]. doi: 10.48550/arXiv.2605.23196.
- Zhu, Y., Kellermann, A., Bowman, D., Li, P., Gupta, A., Danda, A. et al. (2025) 'CVE-Bench: A Benchmark for AI Agents' Ability to Exploit Real-World Web Application Vulnerabilities', *arXiv* [Preprint]. doi: 10.48550/arXiv.2503.17332.
- Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J.Z. and Fredrikson, M. (2023) 'Universal and Transferable Adversarial Attacks on Aligned Language Models', *arXiv* [Preprint]. doi: 10.48550/arXiv.2307.15043.
- Zupic, I. and Čater, T. (2015) 'Bibliometric Methods in Management and Organization', *Organizational Research Methods*, 18(3), pp. 429–472. doi: 10.1177/1094428114562629.