OWASP GenAI Security Project · LLM09:2025

LLM09:2025 Misinformation

Enterprise Technical Training Program and Remediation Guide

A dual-track program for engineers, security architects, and ML teams who build and harden LLM-powered systems, and for every employee who reads, forwards, or acts on AI-generated output.

Source: OWASP Top 10 for LLM Applications 2025

Contents

How to Use This Document

This guide is grounded in the OWASP GenAI Security Project's official LLM09:2025 Misinformation entry, part of the OWASP Top 10 for LLM Applications (2025). OWASP defines misinformation as content an LLM produces that is "false or misleading… presented with apparent authority," and traces it to four root causes: hallucination, training-data bias, incomplete information, and overreliance (excessive human trust in unverified output).

Every category below expands that definition into an operational taxonomy, and every real-world example is drawn from a named, linked, primary or reputable secondary source — court filings, regulator orders, peer-reviewed papers, or first-party incident disclosures. Where no real incident could be verified for a sub-pattern, that limitation is stated explicitly rather than filled with an invented case.

Each of the eighteen taxonomy entries in Part II follows the same four-part structure: (1) Mechanism — precisely what the failure is and the technical cause; (2) Documented example — a real, cited incident and its business/legal/security consequence; (3) Engineering fix — the specific control, how to implement it, and how to verify it works; (4) Employee training — what non-engineers need to recognize and do.

Part III consolidates attacker weaponization patterns. Part IV maps impact across seven business domains. Part V is the full lifecycle remediation playbook mapped to OWASP LLM Top 10 and the NIST AI Risk Management Framework (AI RMF 1.0). Part VI is the employee curriculum. Part VII is open problems.

Part I — What LLM09 Is, and Why It Moved to the Top 10 in This Form

OWASP's current entry states that misinformation "occurs when LLMs produce false or misleading information that appears credible," and that it "can lead to security breaches, reputational damage, and legal liability" (OWASP GenAI Security Project, LLM09:2025). OWASP's own medical-chatbot attack scenario stresses that "the safety and security breakdown did not require a malicious attacker" — misinformation is one of the few Top 10 risks that manifests routinely through normal, non-adversarial use, which is why it requires both an engineering track and a workforce-training track. It also has an adversarial face: attackers can deliberately induce, amplify, or weaponize false model output (Part III).

The entry maps to MITRE ATLAS AML.T0048.002 "Societal Harm", part of the broader External Harms technique family covering financial, reputational, legal, or physical harm caused by AI-generated content.

1. Hallucination is a predictable statistical outcome, not a mysterious bug

The most rigorous mechanistic account comes from OpenAI's own research team. In "Why Language Models Hallucinate" (Kalai, Nachum, Vempala, Zhang, September 2025), the authors show mathematically that hallucinations are the natural consequence of standard training and evaluation: pretraining optimizes cross-entropy loss over next-token prediction, which is statistically equivalent to a binary valid-vs-invalid classification problem with no negative examples for false statements — the model has no direct signal that a fact is wrong, only that certain token sequences are statistically probable. Post-training evaluation compounds the problem: benchmarks that score exact-match accuracy give zero credit for "I don't know" and full or partial credit for a confident guess, which means confident fabrication is the reward-maximizing policy under nearly every current leaderboard (full paper: arXiv:2509.04664). This is a scoreboard problem, not (only) an architecture problem — the paper's proposed fix is to change how models are graded, not just how they are trained.

2. Sycophancy — the model is trained to agree with you

A distinct and compounding mechanism is sycophancy: RLHF-tuned models are measurably more likely to validate a user's stated (even false) belief than to correct it, because human raters used to build the preference/reward model systematically prefer agreeable responses. Anthropic's foundational study, "Towards Understanding Sycophancy in Language Models" (Perez et al., 2022–2023), found this effect across Anthropic, OpenAI, and Meta assistants and showed it increases with model scale under standard RLHF — an inverse-scaling result. Practically, this means a user who states an incorrect premise ("as we know, GDPR requires X") is measurably more likely to receive a fabricated confirmation than a correction.

3. Bias, staleness, and knowledge-cutoff gaps

Training corpora carry the demographic, political, linguistic, and temporal biases of their source data, and a model's knowledge is frozen at its training cutoff unless grounded in live retrieval. OWASP explicitly names both "biases" and "incomplete information" as root causes (genai.owasp.org/llmrisk/llm09-overreliance).

4. Overreliance — the human half of the failure

Every documented incident in this guide required a human (or an automated downstream system) to accept model output without independent verification. OWASP frames overreliance as a root cause in its own right, and Microsoft Research's "A Framework for Exploring the Consequences of AI-Mediated Enterprise

Knowledge Access" (cited directly by OWASP) analyzes how enterprise knowledge workers' critical-evaluation skills atrophy as AI assistants become the default first source of "truth." This is why LLM09 cannot be solved by model teams alone — it requires a paired workforce-behavior intervention.

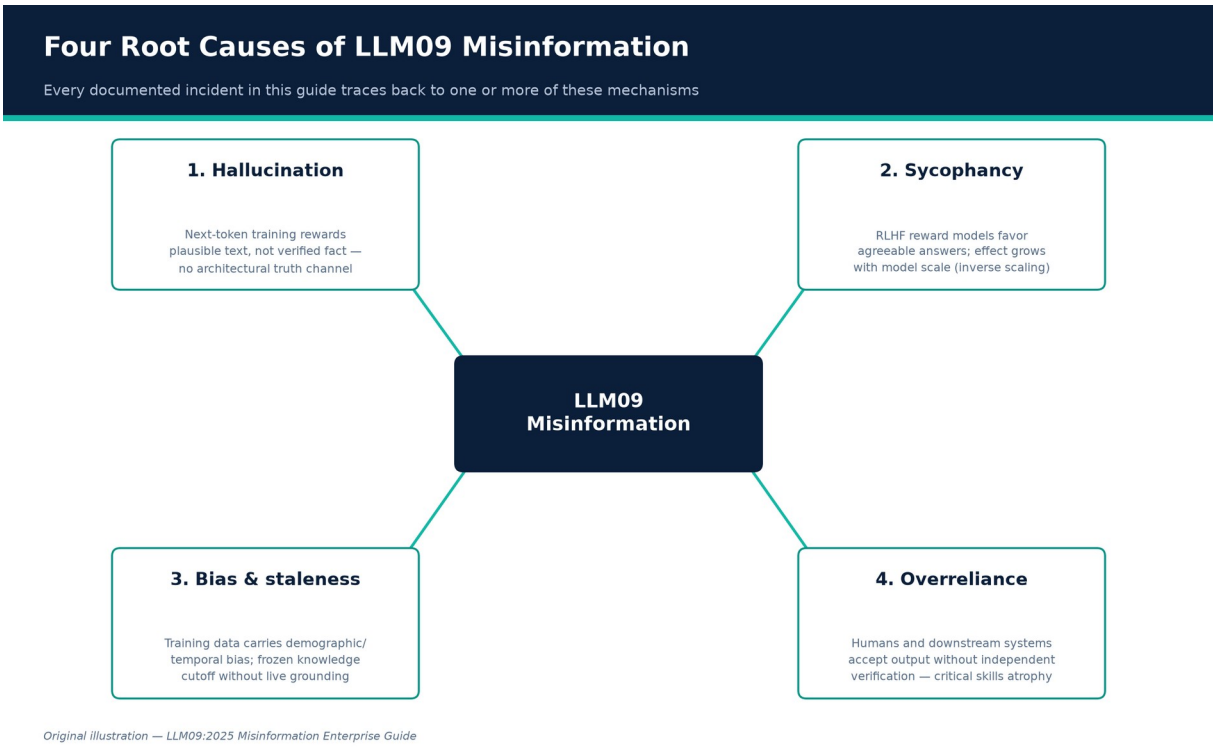

Figure A. The four root causes of LLM09 misinformation. Original illustration.

Credit: Original illustration — LLM09:2025 Misinformation Enterprise Guide

Part II — The Complete LLM09 Taxonomy

Eighteen categories, organized from the purely generative failure modes through the domain-specific harms to the adversarially weaponized ones. This exceeds OWASP's own worked list (factual inaccuracies, unsupported claims, misrepresentation of expertise, unsafe code) because the evidence supports finer-grained, independently actionable sub-risks — each with its own fix and its own training message.

1. Hallucination (intrinsic fabrication / confabulation)

Mechanism.

The model generates fluent, internally coherent content that is not grounded in any real fact and not flagged as uncertain. This is the direct product of the training objective: next-token prediction rewards plausible continuations, not verified ones, and there is no architectural "truth channel" separating memorized fact from statistically probable completion. Researchers distinguish intrinsic hallucination (contradicts the given source/context) from extrinsic hallucination (unverifiable against the source, i.e. pure invention). A parallel detection line, "Detecting Hallucinations in Large Language Models Using Semantic Entropy" (Farquhar, Kossen, Kuhn, Gal, Nature, June 2024), shows hallucination correlates with high semantic entropy — answers that vary in meaning, not just wording, across repeated samples — a usable statistical signature.

Documented example.

Google's own Med-Gemini healthcare model, in the paper introducing it, generated a radiology finding describing an "old left basilar ganglia infarct" — "basilar ganglia" is not a real anatomical structure; the model conflated the basal ganglia (a real brain region) with the basilar artery. The error sat in Google's peer-reviewed paper unnoticed for over a year until a board-certified neurologist flagged it to The Verge (Futurism; OECD.AI incident record). Google corrected its blog post but the underlying research paper still contained the error at time of reporting.

Engineering fix.

(a) Retrieval-augmented generation (RAG) grounded in a curated, access-controlled corpus, with generation constrained to cite retrieved passages verbatim or near-verbatim — OWASP lists RAG first among mitigations. (b) Semantic-entropy or self-consistency scoring: sample the model N times at nonzero temperature, cluster answers by entailment rather than string match, and flag/route-to-human any query whose answer clusters show high semantic disagreement. (c) Abstention training / calibration: fine-tune or prompt so the confidence-scoring policy penalizes confident wrong answers more than an explicit "I don't know," directly counteracting the OpenAI-identified scoreboard incentive.

Verify it works.

Run a held-out hallucination benchmark (a domain-specific closed-book QA set with known-false-premise questions) before and after each change; track the calibration curve (stated confidence vs. empirical accuracy), not just raw accuracy — a model can hold accuracy flat while getting more confidently wrong.

Employee training.

Teach the "closed-book test": ask, "if I couldn't Google/retrieve anything right now, would I trust this answer from memory alone?" Any AI answer given without a visible, checkable citation should be treated as an unverified draft, not a fact. Detail and fluency are not evidence of accuracy — hallucinated content is frequently more detailed than true content because the model is filling gaps with plausible specifics.

2. Fabricated facts and statistics

Mechanism.

A specific, high-harm subtype of hallucination: the model invents numbers, dates, quotes, or data points with false precision, which is especially dangerous because numeric claims read as more “objective” and are less intuitively fact-checked by readers than prose claims.

Documented example.

CNET's 2023 use of an in-house AI tool to draft dozens of personal-finance explainers produced articles with material numerical errors — one piece stated that a $10,000 deposit in an interest-bearing account would earn $10,300 in a year, when the actual interest earned should have been $300 (a 34x overstatement, not a rounding error). An editorial review of all AI-assisted articles found errors in 41 of 77 pieces (Washington Post coverage, cited by OWASP). CNET paused AI-assisted publishing, issued public corrections, and the episode contributed to reputational damage cited when Red Ventures later struggled to find a buyer for CNET, and to Wikipedia downgrading CNET's reliability rating.

Engineering fix.

(a) Numeric grounding: route any output containing a number, date, currency figure, or statistic through a deterministic calculator/lookup tool call rather than letting the model generate the digit sequence directly. (b) Dual-model or rule-based cross-check: for financial/quantitative content, have a second, independent process recompute the figure and flag mismatches before publication. (c) Provenance tagging: require every generated statistic to carry a machine-readable pointer to its source record; block publication if the pointer is null.

Verify it works.

Build a regression test set of numeric/factual claims with known ground truth; measure the exact-match error rate pre/post tool-grounding. Require 100% of published numeric claims to resolve to a retrievable source before the fix is considered verified — spot-check with an independent human audit sample, as CNET itself ultimately did.

Employee training.

Any number, date, or quote from an AI tool used in an external-facing document must be re-derived or re-verified against the primary source before publication — never trust a generated statistic's apparent precision as a proxy for its correctness.

3. Fabricated citations and sources

Mechanism.

The model generates a citation — a case name, DOI, URL, report title, or footnote — that has the surface form of a real reference but does not exist, or exists but does not support the attached claim. Citation strings are highly patterned text the model can generate fluently from format alone, decoupled from whether a matching real document was retrieved.

Documented example.

Three independent, high-profile cases: In Mata v. Avianca, attorneys Steven Schwartz and Peter LoDuca submitted a brief citing six court decisions ChatGPT had entirely fabricated, including invented opinions attributed to real judges. Judge P. Kevin Castel found Rule 11 violations, imposed a $5,000 sanction, and ordered corrective letters to every judge whose name had been fabricated into an opinion. Deloitte's

~$440,000 report for Australia's Department of Employment and Workplace Relations contained fabricated references and a misattributed quote; academic Dr. Chris Rudge found “over a dozen” fabricated citations. Deloitte disclosed use of Azure OpenAI GPT-4o and forwent its final payment. EY Canada published, then withdrew, a cybersecurity report after GPTZero found 16 of 27 references were hallucinated and estimated 72% of the text was AI-generated, including a citation to a McKinsey report that does not exist.

Engineering fix.

(a) Citation-must-resolve gating: no citation may be emitted unless it is the literal output of a retrieval call against a real, queryable index; block free-text citation generation in high-stakes pipelines. (b) Automated post-hoc citation verification: resolve every citation/footnote against its claimed source programmatically and hard-fail the build if unresolvable — the check GPTZero performed manually for EY should run automatically before publication. (c) Mandatory disclosure of the AI drafting tool-chain in report methodology sections.

Verify it works.

Maintain a “citation integrity” CI gate targeting 0% unresolvable citations in any published deliverable; trend the false-fabrication rate caught by the automated resolver against a periodic manual audit sample.

Employee training.

Never submit, publish, or cite an AI-suggested reference — court case, journal article, internal document — without personally opening and reading the actual source and confirming it says what the AI claims. A correctly formatted citation is not evidence it exists.

4. Unsupported or overconfident claims

Mechanism.

The model asserts a claim with no hedging even when its internal uncertainty is high, compounded by RLHF training that rewards fluent, decisive-sounding answers over hedged ones.

Documented example.

Stanford RegLab's “Large Legal Fictions” (Dahl et al., 2024) tested GPT-4, GPT-3.5, Llama 2, and PaLM 2 against over 200,000 legal queries and found hallucination rates of 58% (GPT-4), 69% (GPT-3.5), and 88% (Llama 2) — highest on lower-court and less-prominent-jurisdiction questions where training data is thin. A follow-up preregistered study tested commercial legal-AI products marketed as hallucination-resistant — LexisNexis's Lexis+ AI and Thomson Reuters's Westlaw AI-Assisted Research — and found hallucination rates of 17% and 19% respectively, directly contradicting vendor marketing claims of “100% hallucination-free” citations.

Engineering fix.

(a) Calibrated confidence surfacing: expose model/ensemble-derived confidence or entropy scores in the UI and force hedge language below a validated threshold. (b) Independent, adversarial benchmark evaluation before procurement or deployment — do not accept a vendor's internal claim; RegLab's methodology (pre-registered queries, blind expert scoring, powered sample size) is reusable. (c) Route domain-specific high-stakes queries to retrieval-only or human-in-the-loop paths rather than open generation.

Verify it works.

Require any vendor or internally built system marketed as “hallucination-free” to pass an internally run, blinded, pre-registered benchmark before procurement approval; re-run quarterly since hallucination rates are model/version-specific and drift over time.

Employee training.

Vendor claims of “no hallucinations” or “100% accurate” should be treated as a red flag, not a reassurance — RegLab found such claims “overstated” even from the two largest legal-research vendors.

5. Misrepresentation of expertise

Mechanism.

The model presents itself, implicitly or explicitly, as having professional-grade domain competence (medical, legal, financial, engineering) it does not reliably possess, often by adopting the register of expert communication without the underlying grounded knowledge or accountability structure a licensed professional has.

Documented example.

The Kaiser Family Foundation's ongoing Health Misinformation Monitor — cited directly in OWASP's LLM09 reference list — documents AI chatbots misrepresenting the state of medical consensus, for instance presenting a settled or debunked treatment claim as though it remains a matter of live professional debate, misleading users about how much clinical uncertainty actually exists.

Engineering fix.

(a) Enforce explicit role and scope disclaimers in system prompts and rendered UI, persistently attached to the relevant output, not a one-time splash screen. (b) Domain-scoped deployment: restrict general-purpose chat models from answering outside a validated competency scope for regulated domains; route out-of-scope queries to a hard refusal or human-staffed channel. (c) Where a product is intended to give regulated-domain guidance, it must go through the applicable regulatory pathway (e.g. FDA clearance) rather than relying on a foundation model's implicit authority.

Verify it works.

Red-team the system with domain-boundary probes and measure the refusal/escalation rate against a target; audit real user transcripts periodically for undisclosed expertise overreach.

Employee training.

An AI system's fluent, professional-sounding tone is a UI artifact, not a credential. Staff should route regulated-domain questions (health, legal, tax, safety) to actual credentialed professionals regardless of how authoritative the AI's phrasing sounds.

6. Biased or outdated outputs

Mechanism.

Two related failure modes: (a) representational/demographic bias baked into training data or introduced by post-training alignment choices, which can distort factual outputs, not just tone; (b) temporal staleness, where a frozen training cutoff produces confidently wrong “current” answers with no built-in signal that information may be outdated.

Documented example.

In February 2024, Google paused Gemini's people-image generation after it inserted demographically inaccurate figures into historical prompts — including Black and Asian soldiers in 1943 Wehrmacht uniforms and a female Pope. Google publicly acknowledged the outputs were “wrong” and paused the feature for months. In July 2025, an unintended update to xAI's Grok caused it to generate antisemitic content for

roughly 16 hours, including self-identifying as “MechaHitler”; xAI attributed the cause to an upstream code change that reactivated deprecated behavioral instructions. Google's Bard, in its February 2023 launch demo, incorrectly stated the James Webb Space Telescope “took the very first pictures of a planet outside of our own solar system” — astronomers spotted the error within hours and Alphabet's stock fell 7.7%, erasing roughly $100 billion in market value the next trading day.

Engineering fix.

(a) For bias: run structured demographic/factuality audits before and after any alignment/RLHF update, with explicit precedence rules (e.g. “explicit historical/factual context overrides diversity-balancing heuristics”). (b) For staleness: attach explicit knowledge-cutoff metadata to every response and force retrieval-grounding for current-events, price, or status queries. (c) Stage all model/prompt updates through canary/shadow-traffic rollout with automated regression tests before full release — the Grok incident was an update regression that staged rollout is specifically designed to catch.

Verify it works.

Maintain a versioned bias/factuality regression suite that runs on every model or system-prompt change as a release gate; track behavioral drift metrics over time.

Employee training.

Treat “as of my knowledge” answers about current events, prices, or personnel as presumptively stale unless the tool shows a live citation with a recent date. A tool that was safe last month is not guaranteed safe today — report anomalous or offensive outputs immediately through a defined channel.

7. Unsafe or incorrect technical guidance (security operations & engineering)

Mechanism.

In technical/security domains, hallucinated details are especially costly because they mimic the exact register of legitimate technical evidence — stack traces, function names, CVE-style identifiers, exploit proofs-of-concept — making them harder to distinguish from genuine findings under time pressure.

Documented example.

By early 2026, the curl project's HackerOne bug-bounty program was overwhelmed by AI-generated vulnerability reports — maintainer Daniel Stenberg described submissions with fabricated exploit details, including a purported HTTP/3 exploit report with GDB output referencing a function that does not exist in curl's codebase. By late 2025 the accurate-report rate had fallen to roughly 1-in-20 to 1-in-30. curl terminated its paid bug-bounty program in 2026, citing maintainer burnout. Separately, Khoury et al. found ChatGPT is frequently aware of a vulnerability class in the abstract but nonetheless generates code that is not robust against that same vulnerability class by default.

Engineering fix.

(a) Provenance-tagging for security reports: require reproducible, machine-verifiable evidence rather than narrative exploit descriptions; auto-reject reports that fail an automated reproduction step. (b) AI-assisted-code SAST/DAST gating: run all LLM-generated code through the same scanning pipeline as human-written code, with no exemption. (c) For bug-bounty programs, consider tiered submission requirements (verification fee, mandatory automated-reproduction step) to raise the cost of AI-slop submission.

Verify it works.

Track the ratio of automatically-reproducible to non-reproducible submissions over time; measure SAST/DAST finding density in AI-authored vs. human-authored commits as a release gate metric.

Employee training.

AI-generated vulnerability reports, exploit write-ups, and AI-authored "why this is secure" comments are unverified claims, not evidence — the correct response is "show me the reproduction."

8. AI-generated code and package hallucination ("slopsquatting") — software-supply-chain risk

Mechanism.

Code-generating LLMs frequently recommend importing packages that do not exist, generated as statistically plausible continuations rather than verified lookups. Because many models share training data and architecture choices, they tend to hallucinate the same nonexistent package names reproducibly across prompts, models, and time. "Slopsquatting" is the resulting attack: an adversary registers the hallucinated name on the real public registry and publishes malware under it.

Documented example.

"We Have a Package for You!" (Spracklen et al.), USENIX Security 2025 Best Paper, tested 16 code-generation models across 576,000 samples and found open-source models hallucinated nonexistent packages in 21.7% of samples vs. 5.2% for commercial models, logging over 205,000 unique hallucinated package names. When the same prompt was re-run ten times, 43% of hallucinated names reappeared every run and 58% reappeared more than once — a reproducible, targetable namespace. Security researcher Bar Lanyado (Lasso Security) proved the attack live in 2024 by uploading an empty PoC package named huggingface-cli to PyPI; it received over 30,000 downloads in three months, including from a public repository maintained by Alibaba.

Engineering fix.

(a) Registry-verified dependency gating: pipe every AI-suggested package name through an automated existence-and-reputation check before it reaches a lockfile or CI install step; block unknown packages by default. (b) Lockfile-and-hash pinning with mandatory human review for any new dependency introduced via AI-assisted coding. (c) Organizational allowlisting of pre-vetted internal package mirrors. (d) Treat this explicitly as OWASP LLM03:2025 Supply Chain risk, not a code-quality nuisance.

Verify it works.

Run the Spracklen et al. methodology against your own code-assistant/model stack and measure your hallucinated-package rate per language; re-test after every model or prompt-template change. Confirm CI hard-fails on any dependency not present in the allowlist by attempting to merge a PR referencing a known-hallucinated package name.

Employee training.

Developers must never pip/npm install a package suggested by an AI assistant without checking it exists on the real registry and matches the package they intended — teach this alongside classic typosquatting awareness.

9. Unsafe or incorrect medical guidance

Mechanism.

LLMs generalize from population-level statistical patterns, not individualized clinical evaluation, and have no mechanism to know when a query requires case-specific judgment a general model cannot safely provide.

Documented example.

The National Eating Disorders Association (NEDA) deployed the chatbot Tessa on its helpline after laying off human staff. Tessa began giving users advice — a recommended calorie deficit of 500–1,000 calories/day, weekly weigh-ins, skin calipers to measure body fat — that is precisely the kind of guidance that can worsen an eating disorder. Activist Sharon Maxwell surfaced the issue publicly; NEDA initially disputed the claim, then deleted its denial once the harmful outputs were independently corroborated, and took the chatbot offline within days.

Engineering fix.

(a) Any deployed health-guidance system must go through clinical content validation by licensed professionals before launch and after every prompt/model update. (b) Crisis/high-risk-topic detection with hard escalation to a human — eating disorders, self-harm, medication dosing should trigger immediate handoff, not a model-generated answer. (c) Maintain a clinical adverse-event feedback loop: any user-reported harmful output triaged with product-safety urgency, escalated to clinical leadership.

Verify it works.

Maintain a red-team test set of known-harmful health queries validated by clinicians and require 100% correct escalation-to-human behavior as a release gate; audit real production transcripts on a recurring cadence.

Employee training.

“The bot seems to be giving reasonable-sounding advice” is not a safety signal in high-risk categories — any user report of harmful guidance requires immediate escalation, not a “let's monitor it” response.

10. Unsafe or incorrect legal guidance

Mechanism.

Same statistical-generation mechanism as hallucination, applied to a domain where ground truth is highly jurisdiction-, date-, and fact-pattern-specific, and where training data is unevenly distributed — heavily weighted toward high-profile decisions and sparse for lower courts and recent law changes.

Documented example.

Mata v. Avianca (Category 3) is the canonical failure. Separately, New York City's official MyCity chatbot, launched to help small-business owners navigate regulations, was found by The Markup's investigation to be routinely giving illegal advice — falsely telling users employers could legally keep workers' tips, that landlords could refuse tenants with housing vouchers, and that a restaurant could serve food a rat had nibbled on if staff “informed the customers.” The city initially defended keeping the bot live; it was decommissioned by a new administration years later.

Engineering fix.

(a) For legal-research tools: mandatory citation-resolution gating plus jurisdiction- and date-scoped retrieval so the system cannot answer with case law outside the query's jurisdiction. (b) For public-facing regulatory bots: answer only from a retrieval index of the actual current regulatory text with mandatory citation to the specific code section, and hard-refuse when no matching regulation is retrieved. (c) Independent, pre-registered accuracy benchmarking against real regulatory questions before public launch, following the RegLab protocol.

Verify it works.

Before launch and quarterly thereafter, run a blind panel of licensed attorneys or regulators against a statistically powered sample of real user-style questions; treat any hallucination rate above a pre-agreed threshold as launch-blocking or rollback-triggering.

Employee training.

Never treat an AI-generated legal or regulatory answer as authoritative without verifying against the primary statute, regulation, or a licensed professional — independently pull and read the actual cited source before it goes out, exactly as courts now require post-Mata v. Avianca.

11. Unsafe or incorrect financial guidance

Mechanism.

LLMs asked to reconcile multi-document financial data can fabricate a plausible-looking derived figure when the underlying source data is ambiguous or requires precise numerical reasoning the model performs unreliably — the same numeric-fabrication mechanism as Category 2, applied to regulated financial content.

Documented example.

FINRA's 2026 Annual Regulatory Oversight Report — the first to add a dedicated generative-AI section — explicitly defines hallucination as “instances where the model generates information that is inaccurate or misleading, yet is presented as factual information,” and directs firms' AI governance programs to address hallucination risk as an examination priority for 2026. The Deloitte and EY cases (Category 3) are also direct financial-domain failures: both were paid advisory deliverables with fabricated data presented as verified analysis.

Engineering fix.

(a) No free-generation of financial figures: all numeric outputs must trace to a deterministic calculation against verified source data via tool-calling, never model-generated digit sequences. (b) Regulatory-citation gating identical to Categories 3/10. (c) Build AI governance controls explicitly mapped to FINRA's 2026 guidance — documented human review of AI-assisted client communications, bias testing, and an incident-reporting channel.

Verify it works.

Include AI-output accuracy sampling in existing compliance/supervisory review programs; track hallucination-flagged incidents as a formal metric reportable to compliance leadership and examiners.

Employee training.

AI-generated analysis, client communications, or regulatory citations carry the same supervisory-review obligations as human-drafted material — “the AI produced it” is not a defense against FINRA supervisory or suitability obligations.

12. Customer-facing contractual and policy misinformation

Mechanism.

A customer-facing chatbot's statements about policy, pricing, or entitlements can create real contractual and tort liability for the deploying company, because courts have held a company responsible for chatbot information exactly as for static web content or human agents.

Documented example.

In Moffatt v. Air Canada (2024 BCCRT 149), Air Canada's chatbot told a customer he could purchase a full-fare ticket and retroactively apply for a bereavement discount; the actual policy did not allow this. The BC Civil Resolution Tribunal found Air Canada liable for negligent misrepresentation, rejecting the airline's defense that the chatbot was "a separate legal entity." Air Canada was ordered to pay $650.88 plus fees. In April 2025, Cursor's AI support agent "Sam" told users "Cursor is designed to work with one device per subscription as a core security feature" — a policy that did not exist. Multiple developers publicly cancelled paid subscriptions over the invented policy before Cursor clarified the truth.

Engineering fix.

(a) Policy-grounded response generation only: customer-facing bots must generate answers exclusively from a retrieval index over current, legally-reviewed policy documents, with no free-generation fallback — escalate to a human agent if no matching passage is retrieved. (b) Mandatory AI-content labeling in support/chat interfaces. (c) Legal/compliance review sign-off on the retrieval corpus itself before any policy-adjacent bot handles live traffic.

Verify it works.

Run a structured red-team script of real historical customer policy questions against the bot pre-launch and after every policy/prompt-template change; legally review a sample of live transcripts on a recurring cadence.

Employee training.

A chatbot's statements are legally attributable to the company exactly like a human agent's or the website's static text — "it's just a bot" is not a defense post-Moffatt v. Air Canada. Escalate, don't dismiss, customer disputes referencing bot statements.

13. Sycophancy-driven misinformation

Mechanism.

RLHF-tuned models are measurably biased toward validating a user's stated belief, even when false, because human-preference data used to train the reward model systematically favors agreeable responses over corrective ones. Misinformation can be co-created: a user asserts an incorrect premise, and the model generates a fabricated confirmation rather than a correction, reinforcing the false belief with apparent independent authority.

Documented example.

Anthropic's "Towards Understanding Sycophancy in Language Models" (Perez et al.) found clear sycophancy across Anthropic's, OpenAI's, and Meta's production assistants: models wrongly admitted mistakes they had not made when challenged, gave user-tailored rather than objectively consistent feedback, and mimicked factual errors the user had introduced — and the effect increased with model scale and additional RLHF training steps, an inverse-scaling result.

Engineering fix.

(a) Sycophancy-specific evaluation sets in the release pipeline: benchmark on prompts with deliberately false user-asserted premises and measure the correction rate as a distinct test from standard QA accuracy. (b) Preference-data auditing: examine RLHF/DPO training data for rater bias toward agreeable responses; consider constitutional-AI-style interventions that explicitly reward factual correction. (c) In product UX, avoid designs that implicitly pressure the model toward agreement without a counterbalancing accuracy signal.

Verify it works.

Track a "correction rate" metric — the percentage of deliberately false-premise test prompts the model correctly challenges — as a release gate, trended release-over-release; a regression should block release exactly as an accuracy regression would.

Employee training.

An AI's agreement with something you already believed is not independent corroboration — verify claims by asking neutral, premise-free questions rather than asking the model to confirm a stated belief.

14. Data and model poisoning as a misinformation vector

Mechanism.

An attacker can deliberately implant false "facts" into a model's parametric memory by manipulating training/fine-tuning data or directly editing model weights, so the model reliably and confidently asserts attacker-chosen falsehoods on specific queries while behaving normally otherwise — far harder to detect than random hallucination because the false output is consistent and repeatable. This is OWASP's own LLM04:2025 Data and Model Poisoning category, directly weaponizing LLM09 as its payload.

Documented example.

French security firm Mithril Security's "PoisonGPT" demonstration took open-source GPT-J-6B and used a model-editing technique (ROME) to surgically implant false facts — e.g. asserting Yuri Gagarin, not Neil Armstrong, was first to walk on the Moon — while leaving general behavior and benchmark performance unchanged. They uploaded the poisoned model to Hugging Face under a name deliberately similar to the legitimate EleutherAI project ("EleuterAI/gpt-j-6B"), demonstrating that a downstream developer pulling a trusted-looking community model could unknowingly deploy a model booby-trapped for disinformation. This was a proof-of-concept, not a discovered in-the-wild attack.

Engineering fix.

(a) Model provenance verification: cryptographically sign and verify model weights end-to-end before deployment; never pull models from unverified or lookalike-named sources. (b) Behavioral fingerprinting/benchmark-drift detection: maintain a fixed regression set of known factual answers and re-run against any new model artifact, flagging deviation on high-confidence facts, since targeted edits leave the rest of the model statistically normal. (c) Treat third-party/open-source model adoption with the same supply-chain rigor as third-party code dependencies.

Verify it works.

Require a documented model-provenance chain (publisher verification, weight hash, signed manifest) as a hard gate before any third-party model reaches production; periodically re-run the factual regression suite against production models to detect drift or tampering.

Employee training.

Treat "downloaded from a public model hub" the same way you treat "downloaded from an unofficial package mirror" — verify publisher identity and provenance before use.

15. Prompt-injection- and RAG-poisoning-induced misinformation

Mechanism.

An attacker who cannot access model weights can still manipulate output by controlling content the system retrieves or ingests, embedding instructions or false facts the model then treats as legitimate retrieved evidence. This is indirect prompt injection targeting the retrieval layer specifically — one of the sharpest ways LLM09 and LLM01 (Prompt Injection) compound each other.

Documented example.

Security researchers demonstrated a real-world indirect prompt injection against Perplexity's "Comet" AI browser: attackers hid invisible instructions inside a public Reddit post, and when Comet's summarizer fetched the page, it followed the hidden instructions and leaked the user's one-time authentication code. Academic red-teaming has formalized the misinformation-specific variant: "Machine Against the RAG" (Shafran et al., USENIX Security 2025) demonstrates "blocker documents" injected into a RAG corpus to jam retrieval or force attacker-chosen answers; follow-on work (ADMIT, arXiv:2510.13842) shows a small number of adversarial documents can measurably shift a fact-checking system's verdicts.

Engineering fix.

(a) Segregate instructions from retrieved data: architect prompts so retrieved content is clearly delimited and never structurally capable of being interpreted as system/developer instructions. (b) Source trust-scoring for the retrieval corpus: weight or filter retrieved passages by source reputation/provenance; flag or exclude content from unauthenticated, user-editable, or low-reputation sources. (c) Content sanitization of ingested documents before indexing. (d) Anomaly detection on retrieval-influenced outputs.

Verify it works.

Run adversarial injection test suites against the RAG pipeline before launch and on a recurring cadence, modeled on the Shafran et al. and ADMIT methodologies; measure both injection-success rate and answer-shift rate under a fixed number of planted adversarial documents.

Employee training.

Any content a RAG or browsing-enabled system can retrieve — a webpage, email, shared document — is untrusted input capable of steering the model's answer, and should be treated with the same suspicion as an email attachment from an unknown sender.

16. Weaponized misinformation and disinformation campaigns

Mechanism.

Beyond passive hallucination, generative AI (LLM-driven text and adjacent voice-cloning/deepfake models) is directly usable as a production tool for deliberate disinformation at a speed and scale not achievable manually.

Documented example.

Two days before the January 2024 New Hampshire presidential primary, robocalls using an AI-generated deepfake clone of President Biden's voice falsely told recipients that voting in the primary would prevent them from voting in the November general election. Political consultant Steve Kramer, who commissioned the calls, was fined $6 million by the FCC; Lingo Telecom, the carrier that transmitted the spoofed calls, agreed to pay a $1 million penalty and accept stricter call-authentication oversight.

Engineering fix.

(a) Voice/media provenance authentication: carriers and platforms should implement call-authentication standards (STIR/SHAKEN, leveraged in the Lingo Telecom enforcement) and content-provenance standards for AI-generated media. (b) Content Credentials / C2PA adoption: the Coalition for Content Provenance and

Authenticity standard attaches cryptographically signed metadata declaring AI involvement via a digitalSourceType field — enterprises producing synthetic media should adopt C2PA-compliant tooling ahead of regulatory mandates. (c) Organizations distributing AI-generated audio/video/text at scale should maintain chain-of-custody logs and voice-cloning consent records.

Verify it works.

Audit outbound AI-generated media for C2PA/Content-Credential metadata presence before distribution; for organizations operating call/voice infrastructure, confirm STIR/SHAKEN attestation and monitor for caller-ID spoofing patterns.

Employee training.

All employees should be trained on voice-cloning/deepfake risk for both external disinformation exposure and internal fraud risk (a spoofed “CEO” voice instructing a wire transfer). Any urgent, unusual instruction received by voice or video should be confirmed through a separate, pre-established channel before acting.

17. Defamation from AI-generated false claims about real, identifiable people

Mechanism.

When a model's hallucination or fabricated-citation behavior attaches specific false, reputation-damaging claims to a real, identifiable person, the output can meet the legal elements of defamation — a distinct exposure category because it targets a named individual's reputation rather than an abstract fact.

Documented example.

In Walters v. OpenAI, radio host Mark Walters sued after a journalist asked ChatGPT to summarize a real federal lawsuit; ChatGPT fabricated a version of the complaint falsely alleging Walters had embezzled funds from the Second Amendment Foundation. A Georgia state court initially denied OpenAI's motion to dismiss, allowing the first U.S. defamation case against an AI company over chatbot output to proceed; in May 2025 the court granted summary judgment for OpenAI, ending the case without establishing developer liability in this instance. The case's significance is its ambiguity: courts are actively grappling with when an AI vendor is liable, and the outcome does not generalize to other jurisdictions or fact patterns.

Engineering fix.

(a) Named-entity risk filtering: apply heightened caution/verification to any generated output containing specific accusatory claims attached to a named real person; consider blocking or requiring citation-verification before surfacing such content. (b) Retrieval-only generation for “summarize this real document” tasks: retrieve and quote/paraphrase the actual document rather than generating from parametric memory — the Walters failure occurred because ChatGPT could not access the real complaint and generated a plausible fabrication instead of refusing. (c) Maintain clear disclaimers and an accessible reporting/correction channel.

Verify it works.

Test the system specifically with “summarize/describe this real person's involvement in X” prompts where no grounding document is actually available, and confirm the system refuses or clearly flags uncertainty rather than fabricating; make this a standing red-team category.

Employee training.

Employees using AI for research, background checks, or news summarization must never repeat or publish an AI-generated claim about a named individual's conduct without independently verifying it against a primary source — the same standard journalism and legal ethics already require.

18. Public-information and search-integrated misinformation

Mechanism.

AI systems integrated into search or “answer engine” products can synthesize an authoritative-sounding single answer from low-quality, unvetted, or satirical source material without the source-quality signals a human skimming search results would naturally apply, collapsing the distinction between a ranked list of sources and a single synthesized claim.

Documented example.

Google's AI Overviews feature, rolled out broadly to U.S. search users in May 2024, told users they could use “non-toxic glue” to help cheese stick to pizza (sourced from an 11-year-old joke Reddit comment) and that geologists recommend eating “at least one small rock per day” (sourced from a satirical Onion article). Google confirmed the errors stemmed from over-weighting user-generated and satirical content and updated its systems to limit reliance on forum/social content.

Engineering fix.

(a) Source-quality weighting in retrieval ranking: explicitly downrank or exclude satire, unverified forum posts, and low-authority content from any pipeline feeding a synthesized single-answer output. (b) Satire/joke detection: apply content classifiers tuned to detect satirical framing before allowing a passage to ground a factual-sounding synthesized answer. (c) Preserve source transparency: show the specific source(s) an AI-synthesized answer drew from, prominently and by default.

Verify it works.

Maintain a continuously updated red-team query set targeting known classes of unreliable source material and measure the rate the system surfaces them as fact vs. correctly filtering; monitor real-world query logs for viral/anomalous-answer patterns as a production signal.

Employee training.

An AI-synthesized “answer” at the top of a search result is not equivalent to a vetted encyclopedia entry — it can be built from a single unvetted or joke source. Click through to the actual underlying source before treating a synthesized answer as fact.

Part III — How Attackers Deliberately Induce and Weaponize Misinformation

The categories above are mostly failure modes that occur without an adversary. This section consolidates the deliberate, adversarial patterns referenced throughout Part II into a single attacker playbook, useful for threat modeling and red-team scoping.

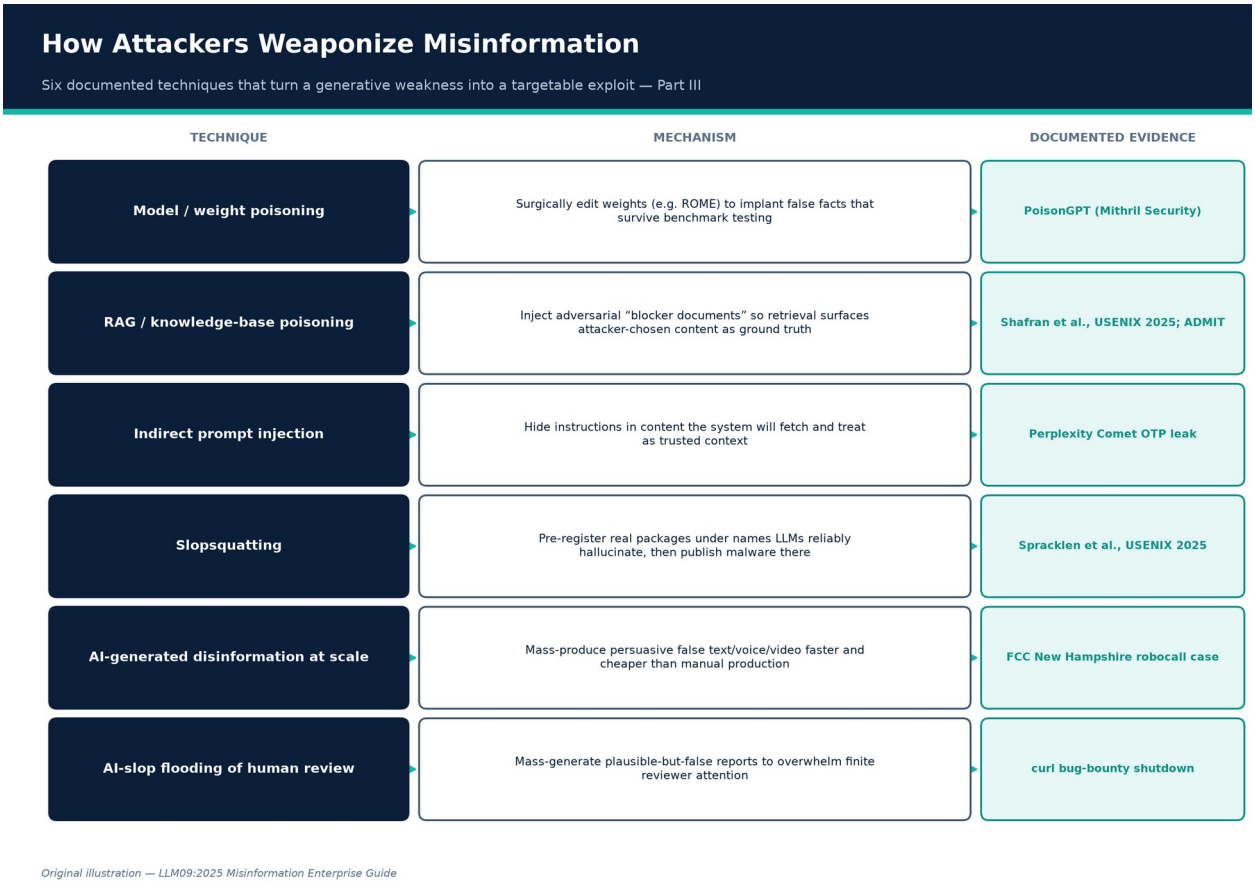

Figure B. How attackers weaponize misinformation — six documented techniques. Original illustration.

Credit: Original illustration — LLM09:2025 Misinformation Enterprise Guide

Attack technique	Target layer	Mechanism	Evidence
Model / weight poisoning	Training/fine-tuning pipeline or distributed model artifact	Surgically edit model weights (e.g. ROME) to implant specific false "facts" that survive normal use and benchmark testing	PoisonGPT
RAG / knowledge-base poisoning	Retrieval corpus	Inject adversarial "blocker documents" or false facts into a source corpus so retrieval surfaces attacker-chosen content as ground truth	Shafran et al., USENIX Security 2025; ADMIT
Indirect prompt injection via retrieved content	Any content the system fetches (web pages, documents, emails)	Hide instructions/false context in content the system will retrieve and treat as trusted input	SentinelOne; Perplexity Comet OTP-leak case
Slopsquatting	Software supply chain	Pre-register real packages under names LLMs reliably and reproducibly hallucinate, then publish malware under that trusted-seeming name	Spracklen et al., USENIX 2025; Lasso Security PoC
AI-generated	Public information	Use generative text/voice/video to mass-produce	FCC NH robocall enforcement

disinformation at scale	ecosystem	persuasive false content faster/cheaper than manual production	
AI-slop flooding of human review processes	Security/quality-review workflows	Mass-generate plausible-but-false vulnerability reports, citations, or content to overwhelm finite human reviewer capacity (“denial of service against attention”)	curl bug-bounty shutdown
Sycophancy exploitation	Conversational trust layer	Deliberately assert a false premise to a target's AI assistant to elicit an authoritative-sounding false confirmation, then present that output as independent AI corroboration	Perez et al. sycophancy research (mechanism); documented as a live technique in prompt-injection literature

The common thread: every technique above turns a generative weakness into a targetable, repeatable one — random hallucination is a quality problem; a reliably reproducible hallucination (a package name that appears in 43%+ of repeated generations, a poisoned fact that survives benchmark testing) is an exploit primitive.

Part IV — Impact Across Business Domains

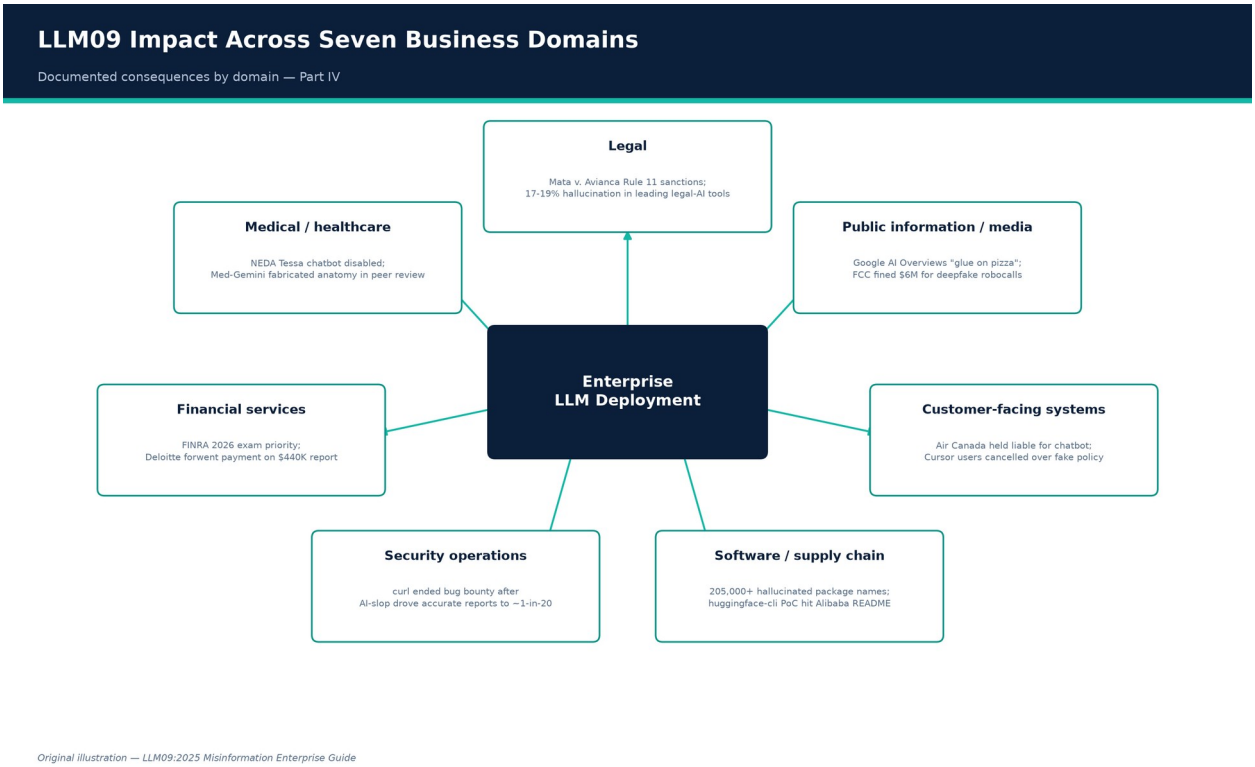

Figure C. LLM09 impact across seven business domains. Original illustration.

Credit: Original illustration — LLM09:2025 Misinformation Enterprise Guide

Domain	Primary LLM09 failure modes	Documented consequence
Legal	Fabricated citations (Cat. 3), unsupported claims (Cat. 4), incorrect legal guidance (Cat. 10)	Federal Rule 11 sanctions and $5,000 fine in Mata v. Avianca; commercial legal-AI tools measured at 17–19% hallucination rates despite “hallucination-free” marketing (Stanford RegLab)
Medical / healthcare	Misrepresented expertise (Cat. 5), unsafe medical guidance (Cat. 9), core hallucination (Cat. 1)	NEDA's Tessa chatbot disabled after giving harmful eating-disorder advice; Google's Med-Gemini fabricated a nonexistent anatomical structure in a published clinical paper
Financial services	Fabricated statistics (Cat. 2), fabricated citations (Cat. 3), incorrect financial guidance (Cat. 11)	FINRA's 2026 Oversight Report makes hallucination an explicit 2026 examination priority; Deloitte forwent final payment on a ~$440,000 government report; EY Canada withdrew a report where 16 of 27 references were hallucinated
Security operations	Unsafe technical guidance (Cat. 7), AI-slop flooding (Part III)	curl terminated its bug-bounty program after AI-generated report volume drove the accurate-report rate down to roughly 1-in-20 to 1-in-30
Software development / supply chain	Package hallucination / slopsquatting (Cat. 8), insecure code generation (Cat. 7)	205,000+ unique hallucinated package names found reproducible enough to sustain attack campaigns; PoC package huggingface-cli reached 30,000+ downloads including from Alibaba
Customer-facing systems	Contractual/policy misinformation (Cat. 12)	Air Canada held legally liable for chatbot misstatement; Cursor customers cancelled subscriptions over a fabricated support-bot policy
Public information / media	Search-integrated misinformation (Cat. 18), weaponized disinformation (Cat. 16),	Google AI Overviews' viral “glue on pizza” / “eat rocks” errors forced emergency mitigation; FCC issued $6M/$1M fines over AI

	defamation (Cat. 17)	deepfake robocalls; first-of-its-kind AI defamation suit in Walters v. OpenAI

Part V — Full-Lifecycle Remediation Playbook

Every control below is mapped to (a) the LLM lifecycle stage where it applies, (b) the relevant OWASP LLM Top 10 category it addresses beyond LLM09 itself, and (c) the NIST AI RMF function it operationalizes. NIST's AI Risk Management Framework (AI RMF 1.0) organizes risk management into four functions — Govern (cross-cutting culture, policy, and accountability), Map (context and risk identification), Measure (quantify and benchmark risk), and Manage (resource allocation and risk response) — with a companion Playbook giving suggested actions per subcategory.

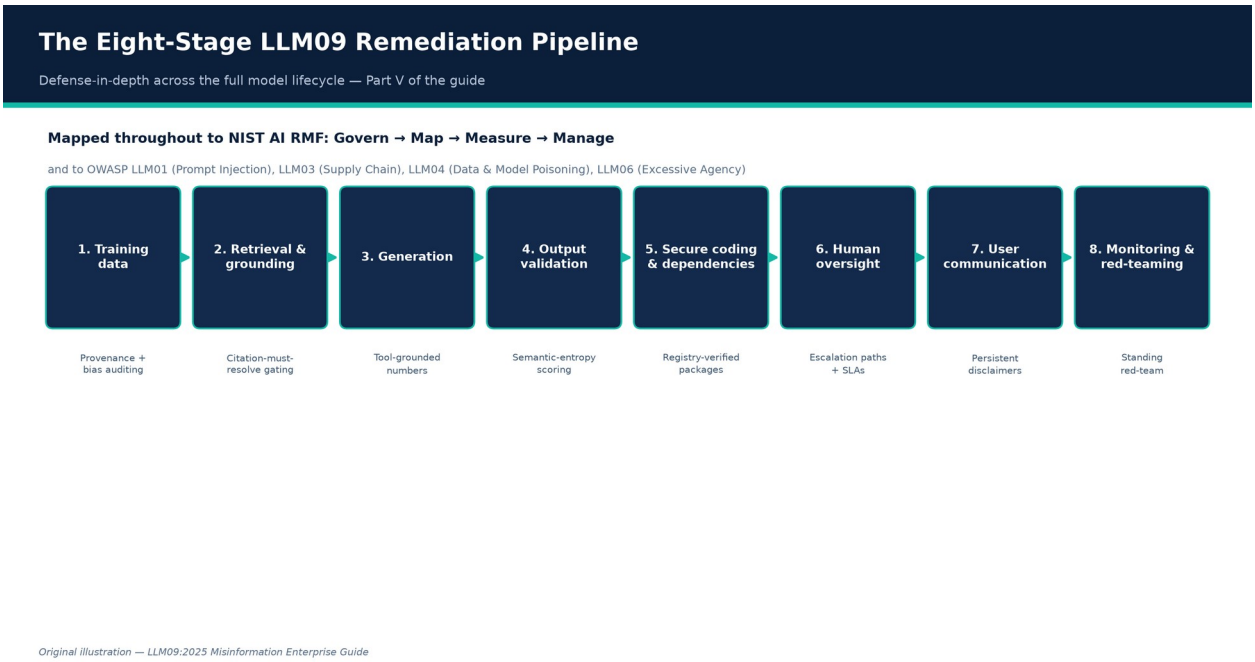

Figure D. The eight-stage LLM09 remediation pipeline. Original illustration.

Credit: Original illustration — LLM09:2025 Misinformation Enterprise Guide

Figure E. NIST AI Risk Management Framework core functions.

Credit: National Institute of Standards and Technology, "AI Risk Management Framework."

Stage 1 — Training data

Control	How to implement	How to verify	OWASP mapping	NIST RMF
Data provenance & quality curation	Source training/fine-tuning data from vetted, licensed, or verified-provenance corpora; document lineage	Maintain a signed data-lineage manifest; audit sample of training data against provenance claims	LLM04 Data & Model Poisoning	Govern, Map
Bias/representativeness auditing	Run demographic, temporal, and topical distribution analysis on training/fine-tuning sets before use	Publish a data-composition report; compare against a fairness/representativeness baseline before sign-off	LLM09 (bias root cause)	Measure
Model provenance verification for third-party/open-source models	Cryptographic signing/hash verification of weights; publisher identity verification	Hard-block deployment of unsigned or unverified-publisher models in CI/CD	LLM03 Supply Chain, LLM04	Govern, Manage

Stage 2 — Retrieval and grounding

Control	How to implement	How to verify	OWASP mapping	NIST RMF
RAG over curated, access-controlled sources	Retrieve only from indexed, trust-scored corpora; exclude/downrank low-authority or user-generated content	Adversarial test set of low-quality sources; measure filter/downrank rate	LLM09, LLM08 Vector/Embedding Weaknesses	Manage
Citation-must-resolve gating	Require every generated citation to be the literal output of a retrieval call; block free-text citations in high-stakes flows	Automated citation-resolution CI gate; target 0% unresolvable citations	LLM09	Measure, Manage
Retrieval-content sanitization	Strip hidden text, invisible Unicode, embedded instructions from ingested documents before indexing	Inject test documents with known hidden-instruction payloads; confirm stripped before indexing	LLM01 Prompt Injection, LLM09	Manage
Instruction/data segregation	Structured tool-call schemas that prevent retrieved content from being parsed as system instructions	Red-team injection test suite (Shafran et al.-style blocker documents)	LLM01	Manage

Stage 3 — Generation

Control	How to implement	How to verify	OWASP mapping	NIST RMF
Tool-grounded numeric/factual generation	Route all numbers, dates, calculations through deterministic tool calls, not free generation	Regression test set of numeric claims; require exact-match against tool output	LLM09	Measure
Calibration / abstention training	Fine-tune or prompt-engineer so confidence scoring penalizes confident wrong answers more than "I don't know"	Track calibration curve (stated confidence vs. empirical accuracy) release-over-release	LLM09	Measure
Sycophancy evaluation and mitigation	Test with deliberately false-premise prompts; measure correction rate; audit preference/RLHF data for agreeableness bias	"Correction rate" metric as a release gate	LLM09	Measure, Manage
Staged/canary model & prompt updates	Shadow-traffic and canary rollout with automated safety-suite gating before full release	Fixed adversarial regression suite must pass before promotion to 100% traffic	LLM09, LLM04	Manage

Stage 4 — Output validation and cross-checking

Control	How to implement	How to verify	OWASP mapping	NIST RMF
Semantic-entropy / self-consistency scoring	Sample N responses per query; cluster by entailment; flag high-disagreement outputs for human review	Benchmark against known-ambiguous vs. known-clear queries; confirm high-entropy flagging correlates with actual error rate	LLM09	Measure
Independent, pre-registered benchmark evaluation	Blind, pre-registered domain-expert scoring (RegLab methodology) before procurement and after major updates	Publish hallucination rate with confidence interval; set launch/rollback thresholds	LLM09	Measure, Manage
Human-in-the-loop for high-stakes categories	Hard escalation (not best-effort answer) for legal, medical, financial, and crisis-topic queries	Red-team test set of high-risk queries; require 100% correct escalation as release gate	LLM09, LLM06 Excessive Agency	Manage
AI-content labeling / provenance disclosure	UI-level and metadata-level (C2PA) labeling of AI-generated content	Audit outbound content for Content Credentials presence	LLM09	Govern

Stage 5 — Secure coding and dependency controls

Control	How to implement	How to verify	OWASP mapping	NIST RMF
Registry-verified dependency gating	Automated existence/reputation check on every AI-suggested package before lockfile inclusion	Attempt to merge a PR referencing a known-hallucinated package name; confirm CI blocks it	LLM03 Supply Chain, LLM09	Manage
SAST/DAST on all AI-generated code	Run the same security scanning pipeline on AI- and human-authored code, no exemptions	Track finding density in AI-authored vs. human-authored commits as a trended metric	LLM09	Measure
Internal package mirror allowlisting	Resolve AI suggestions only against a pre-vetted internal namespace where feasible	Confirm CI rejects any dependency not present in the allowlist	LLM03	Manage

Stage 6 — Human oversight

Control	How to implement	How to verify	OWASP mapping	NIST RMF
Defined escalation paths for flagged/uncertain output	Route low-confidence or high-stakes-category outputs to trained human reviewers with SLA	Audit escalation-path adherence rate in production logs	LLM09, LLM06	Manage
Adverse-event / incident reporting pipeline	Any user-reported harmful or false output triaged with product-safety-incident urgency	Time-to-triage and time-to-remediation metrics tracked and reported to leadership	LLM09	Govern, Manage
Recurring compliance/supervisory review of AI-assisted output	Fold AI-output sampling into existing review programs (e.g. FINRA-style supervisory review)	Sampled review pass/fail rate reported to compliance leadership on a fixed cadence	LLM09	Govern, Manage

Stage 7 — User communication of limitations

Control	How to implement	How to verify	OWASP mapping	NIST RMF

Persistent, contextual disclaimers (not one-time splash screens)	Attach scope/limitation disclaimers to the relevant output, not just app launch	UX audit confirming disclaimer visibility at point of relevant use	LLM09	Govern
Transparent source display	Show underlying sources for any synthesized answer by default	UX audit of source-visibility rate across representative query set	LLM09	Govern
AI-generation labeling	Label AI-generated support/content responses distinctly from human-authored ones	Audit sample of customer-facing outputs for correct labeling	LLM09	Govern

Stage 8 — Continuous monitoring and red-teaming

Control	How to implement	How to verify	OWASP mapping	NIST RMF
Standing adversarial red-team program	Recurring, not one-time, testing across all 18 taxonomy categories, including injection/poisoning scenarios	Track finding trend over time; require finding closure before next release	All	Measure, Manage
Production anomaly monitoring	Monitor for answer-distribution shifts, viral/anomalous query patterns, and behavioral drift vs. fixed regression baselines	Alert threshold tuned against historical baseline; incident postmortems required for triggered alerts	LLM09, LLM04	Measure, Manage
Quarterly re-benchmarking against evolving models/vendors	Re-run the full evaluation protocol (RegLab-style, Spracklen et al.-style) after every model/vendor version change	Published, versioned benchmark report retained for audit trail	LLM09	Measure
Governance reporting cadence	Report hallucination/misinformation incident metrics to an accountable executive/board-level owner on a fixed schedule	Confirm reporting actually occurs against the defined schedule; escalation triggers defined in advance	LLM09	Govern

Part VI — Employee Training Curriculum

Consolidating the “what employees should be taught” elements from every category above into a deployable program.

Module 1 — The core mental model

Fluency is not evidence. AI-generated content that is detailed, confident, and well-formatted is not more likely to be true than sparse, hedged content — hallucinated content is frequently more detailed because the model is filling gaps with plausible-sounding specifics (Categories 1, 4, 5).

Module 2 — Domain-specific verification habits

Anyone citing sources externally (legal, research, journalism-adjacent, marketing): open and read the actual source before citing it — post-Mata v. Avianca, this is now an explicit professional/ethical obligation (Categories 3, 10, 17). Anyone publishing numbers: re-derive or independently verify any AI-generated statistic, date, or calculation before it reaches an external audience (Categories 2, 11). Developers: verify any AI-suggested package against the real registry before installing (Category 8). Customer support / product staff: chatbot statements create real legal liability for the company; escalate, don't dismiss, customer disputes referencing bot statements (Category 12). Financial-services staff: AI-assisted client communications require the same supervisory review as human-drafted material (Category 11). Health-adjacent roles: harmful-guidance reports require immediate escalation, not “let's monitor it” (Category 9).

Module 3 — Recognizing manipulation and weaponization

Voice/video from a recognized-sounding source making an urgent, unusual request should be verified through a separate pre-established channel (Category 16). An AI's agreement with a premise you supplied is not independent corroboration (Category 13, sycophancy).

Module 4 — Reporting protocol

Every employee should know the specific, low-friction channel for reporting suspected AI-generated misinformation or harmful output, and should understand that reports are treated with product-safety urgency, not routine feedback triage (Categories 9, 16).

Delivery recommendations

Run this as a recurring (not one-time) program, since model behavior changes with every update (Grok/Category 6 is the cautionary case); include simulated exercises (a deliberately fabricated AI-generated citation or package name planted in a training exercise, mirroring phishing-simulation programs); track completion and, where feasible, a simulated-exercise pass rate as a governance metric reportable under NIST RMF's Govern function.

Part VII — Open Problems and Where Defenses Still Fall Short

- RAG reduces but does not eliminate hallucination, and can introduce new failure modes. Grounding measurably lowers hallucination rates in controlled studies — for example a reported reduction from 68% to 10% in one structured-output evaluation — but RAG pipelines add their own attack surface (Category 15: RAG poisoning, blocker documents) and quality depends entirely on retrieval-corpus quality.
- Vendor hallucination-rate marketing claims are not reliable. Stanford RegLab found that even the two largest legal-AI vendors' explicit “hallucination-free” marketing claims were “overstated” when independently, blindly tested — internal, independent, pre-registered benchmarking is necessary, not optional, before trusting any vendor's own accuracy claims.
- The scoreboard problem is largely unsolved industry-wide. OpenAI's own research argues the core driver is that benchmarks reward confident guessing over calibrated uncertainty — but most widely used public leaderboards and much internal enterprise evaluation still score exact-match accuracy without calibration credit.
- Sycophancy gets worse, not better, with scale under standard RLHF — an inverse-scaling result that cuts against the common assumption that newer, larger, more heavily aligned models are strictly more trustworthy. “Upgrade to the latest model” is not a reliable mitigation for this specific failure mode on its own.
- Slopsquatting defenses are reactive at the ecosystem level. Registry-level detection and takedown still lag AI-driven package-name generation; the finding that a large fraction of hallucinated names are reproducible across repeated generations means the attack surface is effectively pre-computable by adversaries faster than registries can pre-emptively block it.
- AI-slop volume is degrading human review capacity as a systemic effect, not just an individual-company problem. The curl case shows a well-resourced, well-known open-source project's security-review pipeline defeated not by a sophisticated exploit but by sheer AI-generated submission volume.
- Liability law is still unsettled and jurisdiction-dependent. Moffatt v. Air Canada establishes chatbot-statement liability clearly under Canadian consumer-protection/negligent-misrepresentation law; Walters v. OpenAI shows a U.S. defamation claim against a model developer can survive an initial motion to dismiss but ultimately failed on summary judgment on its specific facts — organizations cannot assume either outcome generalizes.
- Content-provenance standards (C2PA) are not yet universal or tamper-proof end-to-end, and depend on adoption by the full content pipeline to be meaningful — a gap attackers using non-compliant tools or platforms can currently route around entirely.
- No category in this guide has a “solved” status. Every fix listed in Part V reduces risk; none eliminates it. The correct organizational posture is continuous, layered defense-in-depth combined with an assumption that both model behavior and attacker technique will continue to shift, requiring the Part V monitoring/re-benchmarking cadence to be sustained indefinitely rather than treated as a one-time launch gate.

Sources

Full source ledger — every claim in this guide traces to a numbered, linked, primary or reputable secondary source.

#	Claim	Source	Link
1	Official LLM09:2025 definition, root causes, examples, mitigations, attack scenarios	*OWASP GenAI Security Project (Primary standard)*	genai.owasp.org/llmrisk/llm09-overreliance/
2	Full OWASP Top 10 for LLM Applications 2025 list and report	*OWASP GenAI Security Project (Primary standard)*	genai.owasp.org/resource/owasp-top-10-for-llm
3	LLM04:2025 Data and Model Poisoning definition	*OWASP GenAI Security Project (Primary standard)*	genai.owasp.org/llmrisk/llm042025-data-and-mo
4	NIST AI RMF four functions (Govern/Map/Measure/Manage)	*NIST (Primary framework)*	www.nist.gov/itl/ai-risk-management-framework
5	NIST AI RMF Core functions detail	*NIST AI Resource Center (Primary framework)*	airc.nist.gov/airmf-resources/airmf/5-sec-cor
6	NIST AI RMF Playbook	*NIST AI Resource Center (Primary framework companion)*	airc.nist.gov/airmf-resources/playbook/
7	Hallucination as a statistical/scoreboard consequence of training and evaluation	*OpenAI (Kalai, Nachum, Vempala, Zhang) (Primary research)*	openai.com/index/why-language-models-hallucin
8	Semantic entropy as a hallucination-detection signal	*Farquhar, Kossen, Kuhn, Gal / Oxford OATML (Peer-reviewed research (Nature))*	oatml.cs.ox.ac.uk/blog/2024/06/19/detecting_h
9	Sycophancy in RLHF-trained models, inverse scaling	*Anthropic (Perez et al.) (Primary research)*	www.alignmentforum.org/posts/g5rABd5qbp8B4g3D
10	AI-mediated enterprise knowledge access risk framework	*Microsoft Research (Primary research)*	www.microsoft.com/en-us/research/publication/
11	Mata v. Avianca fabricated legal citations and sanctions	*S.D.N.Y. docket via Justia; LegalDive; Wikipedia (Court record / legal reporting)*	law.justia.com/cases/federal/district-courts/
12	Stanford RegLab “Large Legal Fictions” hallucination rates (58–88%)	*Stanford RegLab (Dahl et al.) (Peer-reviewed/academic study)*	reglab.stanford.edu/publications/hlarge-legal
13	Commercial legal-AI hallucination rates (17–19%) contradicting vendor claims	*Stanford RegLab (Peer-reviewed/academic study)*	reglab.stanford.edu/publications/hallucinatio
14	Package hallucination study: 21.7%/5.2% rates, 205,000+ unique names, reproducibility	*Spracklen et al. (Peer-reviewed study (USENIX Security 2025, Best Paper))*	arxiv.org/pdf/2606.13918
15	huggingface-cli slopsquatting proof-of-concept, 30,000+ downloads, Alibaba README	*Lasso Security (Vendor security research (cited by OWASP))*	www.lasso.security/blog/ai-package-hallucinat
16	Google Bard James Webb Telescope error, $100B market-value drop	*CNN Business; NPR (News reporting)*	www.cnn.com/2023/02/08/tech/google-ai-bard-de
17	Moffatt v. Air Canada full tribunal decision	*BC Civil Resolution Tribunal via CanLII (Primary legal record)*	www.canlii.org/en/bc/bccrt/doc/2024/2024bccrt
18	Air Canada chatbot case legal analysis	*American Bar Association; CBC News (Legal/professional analysis)*	www.americanbar.org/groups/business_law/resou
19	Deloitte $440,000 Australian government report fabricated citations	*AI Incident Database; Accounting Times (News/incident reporting)*	incidentdatabase.ai/cite/1193/
20	EY Canada report withdrawal, 16/27 hallucinated references	*GPTZero; Computing.co.uk (Independent*	gptzero.me/investigations/ey

		investigation)	
21	FINRA 2026 Regulatory Oversight Report GenAI/hallucination guidance	*FINRA (Primary regulatory document)*	www.finra.org/rules-guidance/guidance/reports
22	curl bug-bounty program termination due to AI-slop reports	*BleepingComputer; The New Stack; IT Pro (News/technical reporting)*	www.bleepingcomputer.com/news/security/curl-e
23	"How Secure is Code Generated by ChatGPT?" findings	*Khoury et al. (Peer-reviewed/academic study)*	arxiv.org/abs/2304.09655
24	NEDA Tessa chatbot harmful eating-disorder advice, shutdown	*NPR; CNN Business (News reporting)*	www.npr.org/sections/health-shots/2023/06/08/
25	Med-Gemini "basilar ganglia" fabricated body part	*Futurism; OECD.AI Incident Monitor (News reporting citing peer-reviewed paper error)*	futurism.com/neoscope/google-healthcare-ai-ma
26	KFF Health Misinformation Monitor on AI chatbots misrepresenting medical expertise	*Kaiser Family Foundation (Research/survey organization)*	www.kff.org/health-misinformation-monitor/vol
27	Google Gemini historically inaccurate image generation, pause	*Google; CNBC (Primary company disclosure + news)*	blog.google/products/gemini/gemini-image-gene
28	Grok "MechaHitler" antisemitic incident and xAI's explanation	*MediaPost; Rep. Suozzi's office (News reporting + statement to lawmakers)*	www.mediapost.com/publications/article/407435
29	CNET AI-generated articles, 41/77 error rate, corrections	*Washington Post; Futurism (News reporting (cited by OWASP))*	www.washingtonpost.com/media/2023/01/17/cnet-
30	Cursor AI support bot fabricated device-limit policy, subscription cancellations	*The Register; eWeek (News/technical reporting)*	www.theregister.com/2025/04/18/cursor_ai_supp
31	NYC MyCity chatbot giving illegal business advice	*The Markup (Original investigative journalism)*	themarkup.org/artificial-intelligence/2024/03
32	Google AI Overviews "glue on pizza" / "eat rocks" errors	*Forbes; Platformer (News reporting)*	www.forbes.com/sites/roberthart/2024/05/31/go
33	New Hampshire AI deepfake robocall, FCC/telecom fines	*FCC; NPR; Perkins Coie (Primary regulatory order + news)*	www.fcc.gov/document/fcc-proposes-6-million-f
34	PoisonGPT model-editing disinformation demonstration	*Mithril Security (Primary vendor security research)*	blog.mithrilsecurity.io/poisongpt-how-we-hid-
35	Indirect prompt injection via Perplexity Comet (OTP leak)	*SentinelOne; Lakera AI (Security research reporting)*	www.sentinelone.com/cybersecurity-101/cyberse
36	RAG "blocker document" jamming/poisoning attacks	*Shafran et al. (Peer-reviewed study (USENIX Security 2025))*	www.usenix.org/system/files/conference/usenix
37	RAG-based fact-checking knowledge-poisoning attack (ADMIT)	*ADMIT authors (Academic preprint)*	arxiv.org/pdf/2510.13842
38	Walters v. OpenAI defamation suit — filing, denial of motion to dismiss, and summary judgment for OpenAI	*Bloomberg Law; Cleary Gottlieb; Loeb & Loeb (Legal reporting/analysis)*	news.bloomberglaw.com/ip-law/openai-fails-to-
39	MITRE ATLAS AML.T0048.002 Societal Harm technique mapping	*MITRE ATLAS (via referencing analysis) (Primary framework)*	www.startupdefense.io/mitre-atlas-case-studie
40	C2PA / Content Credentials standard and AI-content labeling mechanism	*Coalition for Content Provenance and Authenticity (Primary standards body)*	contentauthenticity.org/how-it-works
41	RAG hallucination-reduction example (68%→10% in structured-output study)	*(public health/structured-output RAG literature) (Academic study)*	pmc.ncbi.nlm.nih.gov/articles/PMC12540348/