Original Scientific Paper/Original naučni rad
Paper Submitted/Rad primljen:31.12.2025.
Paper Accepted/Rad prihvaćen:10.01.2026.
DOI: 10.5937/SJEM2601077L

UDC/UDK: 005.92:343.452
004.65:004.8

Employee-Driven Leakage of Technical Documentation into General-Purpose LLMs: An Integrative Review

Luka Latinović[1], Oleg Zhukovskiy[2], Olga Mašić[3], Dejan Živković[4]

[1]Belgrade School of Engineering Management, Beopolis University, Serbia, luka.latinovic@fim.rs
[2]MART-INFO LLC (ООО «МАРТ-ИНФО»), Moscow, Russian Federation
[3] Belgrade School of Engineering Management, Beopolis University, Serbia, olga.masic@fim.rs
[4] [1]Belgrade School of Engineering Management, Beopolis University, Serbia, dejan.zivkovic@fim.rs

Abstract: General-purpose large language models are increasingly used by employees to interpret standards, troubleshoot systems, and draft or refine engineering artefacts. This routine assistance creates bidirectional flows: proprietary documentation is occasionally externalised as prompts, uploads, screenshots, or connector-mediated retrieval, while model outputs are pasted back into internal tickets, runbooks, and repositories. This integrative review synthesises heterogeneous evidence (peer-reviewed research, provider and regulator materials, and structured incident reporting) to map employee-driven leakage mechanisms along the documentation lifecycle and to derive a governance approach that is auditable under policy drift and multi-vendor toolchains. We identify a recurrent set of boundary-crossing transition points such as copy–paste, upload/OCR, connector invocation, and paste-back, where risk concentrates and where observability is often weakest. Across these pathways, four cross-cutting risk dimensions recur: confidentiality and competitive exposure, compliance and cross-border transfer, model-side effects (including extraction, spillover, and contamination risks), and incentive-driven governance gaps that sustain shadow workflows. Building on the mechanism map, we propose a proportionate "minimal guardrail stack" and an organisational evaluation framework combining qualitative risk scoring, rule-based escalation, and simple, trackable metrics (e.g., consolidation onto sanctioned channels, blocking effectiveness, inspection false positives, policy-drift lag, and time to a compliant alternative). The paper does not assert prevalence. Instead, it aims to make assumptions explicit and support cautious, workflow-compatible adoption decisions.

Keywords: data exposure, data exfiltration, training capture, shadow IT, AI governance, model-mediated knowledge transfer, egress.

Neovlašćeno otkrivanje tehničke dokumentacije od strane zaposlenih velikim jezičkim modelima opšte namene: integrativni pregled

Sažetak: Opštenamenski veliki jezički modeli sve se češće koriste među zaposlenima za tumačenje standarda, otklanjanje problema u sistemima i izradu ili doradu inženjerskih artefakata. Ova rutinska pomoć stvara dvosmerne tokove: vlasnička dokumentacija se povremeno iznosi izvan organizacije kroz upite (promptove), otpremanje fajlova, snimke ekrana ili preuzimanje posredstvom konektora, dok se izlazi modela zatim kopiraju nazad u interne tikete, operativna uputstva i repozitorijume. Ovaj integrativni pregled sintetizuje heterogene dokaze (recenzirana istraživanja, materijale pružalaca usluga i regulatora, kao i strukturisane izveštaje o incidentima) kako bi mapirao mehanizme curenja koje pokreću zaposleni duž životnog ciklusa dokumentacije i izveo pristup upravljanju koji je proverljiv u uslovima promena politika i višedobavljačkih lanaca alata. Identifikujemo ponavljajuće prelazne tačke na kojima se prelaze granice poverenja kao što su kopiraj–nalepi, otpremanje/OCR, pozivanje konektora i vraćanje sadržaja kopiranjem, na kojima se rizik koncentriše i gde je mogućnost uočavanja često najslabija. Kroz ove putanje ponavljaju se četiri poprečne dimenzije rizika: poverljivost i konkurentska izloženost, usklađenost i prekogranični prenos, efekti na strani modela (uključujući rizike ekstrakcije, „prelivanja" i kontaminacije) i upravljački jazovi vođeni podsticajima koji održavaju „shadow" tokove rada. Na osnovu mape mehanizama predlažemo proporcionalan „minimalni paket zaštitnih ograda" i organizacioni okvir za evaluaciju koji kombinuje kvalitativno ocenjivanje rizika, eskalaciju zasnovanu na

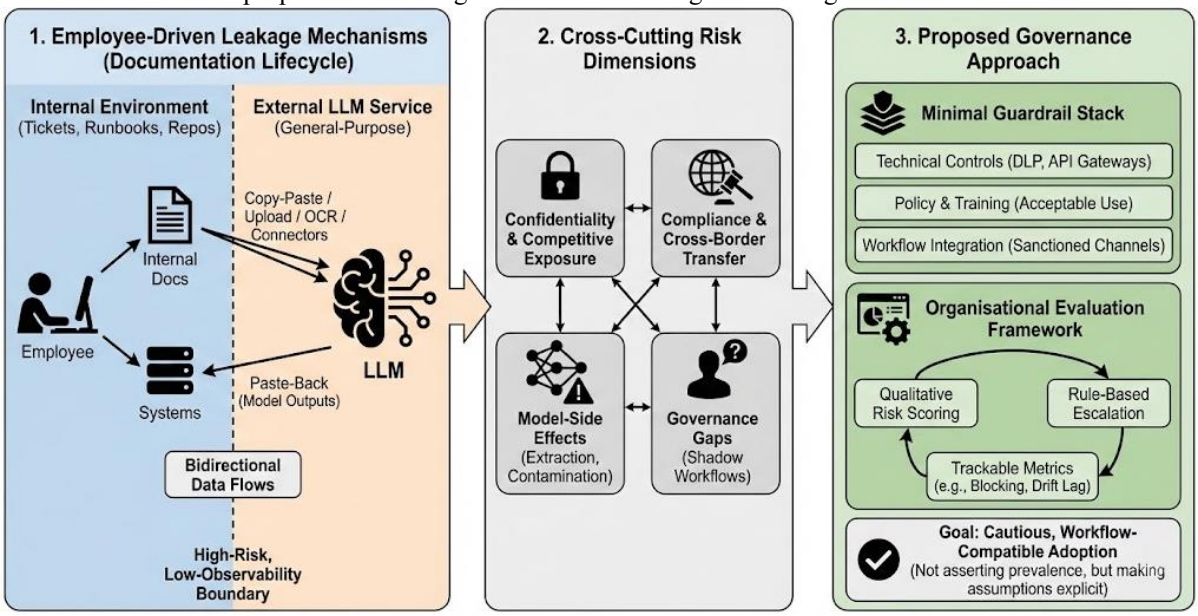
pravilima i jednostavne, pratljive metrike (npr. konsolidaciju na odobrene kanale, efikasnost blokiranja, lažno pozitivne nalaze inspekcije, kašnjenje u usklađivanju sa promenama politika i vreme do usaglašene alternative). Rad ne tvrdi učestalost pojave. Umesto toga, cilj je da pretpostavke učini eksplicitnim i podrži oprezne odluke o usvajanju koje su kompatibilne sa stvarnim tokovima rada.

Ključne reči: izloženost podataka, eksfiltracija podataka, obuhvat za trening, neautorizovani IT, upravljanje VI, modelom posredovan prenos znanja, izlazni tok (egres).

1. Introduction

General-purpose large language models (LLMs) appear to have moved from occasional summarisation and translation into routine problem solving within engineering and documentation work (Saka et al., 2024; J. Yang et al., 2024). Practitioners consult them to interpret standards, draft procedures, refactor build and computer-aided design (CAD)-adjacent scripts, and diagnose integration issues; their suggestions are then edited and incorporated into runbooks, standard operating procedures (SOPs), and specifications (Sajjadi Mohammadabadi et al., 2025). In parallel, employees may paste or upload excerpts of proprietary material to obtain targeted assistance (Malki et al., 2025a). This bidirectional exchange—documentation flowing outward to the model and model outputs flowing back into official artefacts—creates a plausible channel for exposure of technical knowledge to external systems and, conversely, for model-mediated knowledge to diffuse across organisations (Figure **Error! eference source not found.**).

Figure 1: Systematic mapping of employee-driven documentation leakage into general-purpose LLMs, illustrating bidirectional data flows across critical transition points (copy-paste, uploads, connectors) and the proposed "minimal guardrail stack" for organisational governance.



The mechanisms are principally socio-technical. Outbound disclosure can occur through direct prompts, file uploads, and screenshots subjected to optical character recognition (OCR), as well as through intermediaries such as browser extensions, plugins, and third-party chat “wrappers” that relay content to additional processors (Acar et al., 2020). These actions cross trust boundaries at the moments of copy–paste, upload, connector invocation, and paste-back. Provider data-use practices add further uncertainty: operational logging and safety/abuse telemetry may persist even where customer inputs are excluded from parameter updates (Clusmann et al., 2025; Ferrag et al., 2025). For clarity, this review uses the term training capture as operational shorthand for inputs later influencing model or feature improvement under some products or configurations; where inputs are excluded from parameter updates, operational retention (logs, safety traces, analytics) may still be material. Because model parameters are shaped by prior corpora and interactions, responses cannot be cleanly attributed to a single source, so cross-organisational influence is plausible even when any single exchange appears benign.

The stakes are non-trivial. Technical documentation—requirements and design specifications, CAD exports and bill-of-materials (BOM) notes, SOPs and runbooks, test and maintenance logs, configuration and deployment guides—often resides in version control systems (VCS) (Zolkifli et al., 2018), wikis (Standing & Kiniti, 2011), and product/application lifecycle management platforms (PLM/ALM) (Ebert, 2013). These artefacts encode trade secrets, tacit know-how, and occasionally personal data. Leakage may weaken claims to secrecy, complicate intellectual-property positions, or trigger data-transfer and confidentiality obligations (Prinz, 2025). Organisational incentives also matter: where productivity is rewarded and sanctioned alternatives are absent, employees may normalise unsanctioned LLM use, reducing observability at precisely the points where governance is needed (Williams et al., 2025).

2. Methodology

This study adopts an integrative review design, a methodology specifically chosen for its capacity to combine diverse data sources—including peer-reviewed literature, technical white papers, and incident reports—to generate new conceptual frameworks. Unlike systematic reviews that prioritise prevalence estimates, an integrative design focuses on synthesising these heterogeneous streams to conceptualise the leakage mechanisms and derive the governance model proposed in Section 6. Rather than aiming for the exhaustive quantification typical of systematic reviews, this integrative review prioritises conceptual synthesis and mechanism mapping, tracing how specific work practices and infrastructural choices produce employee-driven leakage pathways, while making limitations and assumptions explicit (Torraco, 2005; Whittemore & Knafl, 2005). Inclusion required relevance to employee LLM use, provider handling of user inputs, documentation-relevant risks, or mitigations/governance. Sources were coded by interaction surface (chat/API/plugin), data path (prompt/upload/relay), stated handling (retention/training/telemetry), and controls. Conflicting claims were retained as bounded scenarios. Research questions address pathways, risks, proportionate controls, and organisational evaluation.

Research questions.

- RQ1: Through which employee-driven pathways do proprietary documentation artefacts plausibly reach general-purpose LLM services?
- RQ2: What risk types arise, including confidentiality/IP loss, compliance and cross-border transfer, and model-side mechanisms (spillover, extraction, contamination)?
- RQ3: Which socio-technical controls appear proportionate, contractable, and auditable under realistic organisational incentives?
- RQ4: How should organisations evaluate providers and architectures using a qualitative scoring scheme, rule-based escalation to a minimal guardrail stack, and simple metrics?
- RQ1–RQ3 correspond to the review and synthesis components; RQ4 motivates the derivation of the operational risk-assessment framework.

2.1. Source identification and inclusion

Given the focus on mechanisms rather than prevalence estimates, the corpus intentionally spans:

- Peer-reviewed literature on LLM use in work settings, data leakage, cloud governance, and security/privacy of extensions, APIs, and cloud email/document platforms;
- Provider documentation (e.g. data-use policies, privacy/security white papers, product architecture descriptions) where these affect handling of user inputs and technical artefacts;
- Regulatory, standards, and guidance documents relevant to data protection, cross-border transfers, and safety/criticality of technical information;
- Structured incident databases and well-documented case reports involving generative-AI or adjacent cloud tooling where leakage of work artefacts is plausibly involved.

Inclusion required at least one of the following: (i) empirical or documented examples of employee LLM use in professional contexts; (ii) explicit discussion of how providers handle user inputs (retention, training use, telemetry); (iii) documentation-relevant risks (e.g. code, specifications, logs, design documents); or (iv) concrete mitigations, controls, or governance models that bear on such risks. Opinion pieces and purely speculative commentary without traceable mechanisms were excluded.

Searches were conducted iteratively in major scholarly databases and preprint servers, complemented by backward and forward citation chasing and targeted searches of provider and regulator sites. Given the rapid evolution of tooling and policies, inclusion emphasised recency and relevance to current LLM deployments, with older work drawn in selectively where needed to anchor long-standing mechanisms (e.g. DLP limitations, browser/endpoint artefacts).

2.2. Coding and mechanism-oriented synthesis

Eligible sources were coded using a structured template oriented around interaction surface, data path, handling, and controls:

- Interaction surface: chat interface, API, plug-in/extension, assistant, email/doc integration, CI/CD pipeline, link-sharing, endpoint.
- Data path: prompt, upload, relay, logging/telemetry, sync, or secondary redistribution.
- Stated handling: retention periods, training use, internal sharing, telemetry practices, and any constraints or guarantees.
- Controls: technical safeguards (e.g. DLP, gateways, key management), contractual controls, and organisational practices.

Sources were also annotated for evidence type (empirical study, documented incident, design/architecture description, policy text) and for their primary locus of risk (confidentiality/IP, regulatory/compliance, safety-criticality, or model-side effects). Conflicting claims, e.g., on whether a given provider uses customer inputs for training under specific plans, were retained as bounded scenarios rather than harmonised away: where policies, implementations, or interpretations diverged, this divergence was treated as an empirical feature of the ecosystem.

The synthesis proceeded in three steps. First, coded material was used to trace concrete leakage pathways from employee actions to LLM ecosystems, grouping similar mechanisms into the typology presented in Table 1. Second, across pathways, the review identified systematic misalignments between impact, likelihood, and observability, and recurrent patterns of control failure (e.g. policy-only measures, over-reliance on DNS blocking, lack of OCR-aware inspection, permissive OAuth scopes). Third, these patterns were cross-checked against organisational incentive structures and deployment realities to assess which proposed controls appear feasible and proportionate.

2.3. Derivation of the risk-assessment framework

The qualitative risk-assessment framework (Section 6) is explicitly derived from this mechanism-oriented synthesis rather than introduced as an independent model. For each pathway in the typology, severity and likelihood were assessed on ordinal four-point scales, with an observability modifier capturing detection difficulty and a confidence tag capturing the quality and consistency of the underlying evidence. These judgements are not claimed as precise measurements; instead, they provide a transparent rubric for prioritising governance attention and for making underlying assumptions inspectable.

Building on the identified control patterns, controls were grouped into three deployment tiers: a universal baseline (“Tier 0”), a gate-and-redact layer (“Tier 1”) applied to higher-priority or low-visibility pathways, and an architectural reduction tier (“Tier 2”) reserved for the highest-consequence or regulated documentation classes. Rule-based escalation from the qualitative risk scores to these tiers was then specified in auditable form, reflecting RQ4’s emphasis on proportionate, contractable, and operationally realistic guardrails.

Finally, a small set of simple metrics was defined to allow organisations to track whether the adoption of these guardrails is consolidating LLM use onto governed channels, reducing reliance on unsanctioned tools, and keeping governance responsive to changes in tooling and provider policies. These metrics are intentionally comparative rather than absolute, consistent with the mechanism-oriented focus of this integrative review: they are designed to support internal learning and recalibration, not to claim definitive quantification of LLM risk.

3. Typology of leakage pathways

This section classifies recurrent employee-driven routes by which documentation crosses trust boundaries. Categories overlap and often chain (e.g., screenshot → OCR → plugin relay → paste-back). Likelihood and impact are qualitative, reflecting context (sector, document criticality, provider tier, endpoint hygiene). Crucially, the table pairs each vector with its characteristic control failures and maps them to feasible technical mitigations,

forming the basis for the governance stack in Section 6. Table 1 is intended as a gap-analysis instrument and a living register.

Table 1: Leakage pathways for employee-driven exposure of proprietary technical documentation to general-purpose LLM ecosystems, characterised by immediate mechanism ("vector"), observability, likelihood, impact, typical organisational control failures, and feasible mitigations.

Pathway	Vector (immediate mechanism)	Observability	Likelihood	Impact	Typical control failures	Feasible controls	Example references
Direct prompt copy-paste of proprietary text	Snippets from specs/SOPs/logs pasted into public chat for task help; outputs pasted back into drafts	Often only egress domain; prompt content invisible unless gateway inspects	**High**	Ranges from minor context loss to disclosure of trade-secret parameters	No point-of-use labels; no JIT warnings; policy-only prohibitions	Prompt gateway with content inspection; client-side redaction/templates; sanctioned enterprise chat with retention-off	Incident 768: ChatGPT Implicated in Samsung Data Leak of Source Code and Meeting Notes (*Incident 768*, 2023)
File uploads (docs/archives) and screenshots→ OCR	Whole files or images dragged into chat; OCR extracts text, bypassing text-only DLP	MIME size/type visible; OCR text unseen unless OCR-aware DLP	**Med–High**	Higher than copy-paste (bulk movement)	DLP not OCR-aware; permissive MIME allow-lists; metadata left intact	OCR-aware inspection; strip metadata; sandbox viewers; restrict bulk uploads to sanctioned endpoints	Stealthy Information Leakage from Android Smartphone through Screenshot and OCR (Y. Kim et al., 2015)
Plugins / extensions / third-party "wrappers"	Tools relay prompts/files to additional vendors/ services	Tool call chains opaque; retention varies by tool	**Medium (grows with adoption)**	Amplified by multi-vendor propagation	Unvetted enablement; broad OAuth scopes; no kill-switch	Curated catalogues; per-tool scoping; disable unsanctioned wrappers; provenance logs	(*Incident 1186*, 2025; Starov & Nikiforakis, 2017)
API relays & server-side logging	Internal scripts/apps forward docs to external APIs; gateways/CI proxies log payloads	Network-layer visibility high; verbose logs become sensitive stores	**Medium**	Durable log exposure; developer environment spillage	Keys in code; excessive request/response logging; debug proxies	Secrets managers; minimum-necessary logging with redaction; scoped service accounts	Detecting Misuse of Security APIs (Mousavi et al., 2025)
Email/Doc assistants auto-syncing to provider clouds	Mail/drive assistants ingest repositories for "smart" features	OAuth scopes visible; actual sync/retention opaque tenant-wide	**Medium**	Broad ingestion (mailboxes/ drives)	"Accept all" scopes; weak off-boarding of assistants; unclear residency	Least-privilege OAuth; opt-out by default; residency/retention terms in DPAs	Security challenges for cloud-based email infrastructure (Bhardwaj & Goundar, 2017)
Shared links / chat transcripts / permissive defaults	"Anyone with the link" sharing; exported chat threads; external re-sharing	Some audit trails; hard to track onward sharing	**High in collaborative organisations**	Wide unintended audience; persistent re-disclosure	Public-link defaults; long-lived tokens; no expiry	Domain-restricted links; expiries; secret scanning on shared artefacts	Link-based sharing, "anyone with the link", large unintended audience (Wan et al., 2024)
Endpoint artefacts (clipboard / caches / sync)	Clipboard history, local/browser caches, synced profiles retain prompts/ responses	Low at time of event; surfaces later in support/legal	**Medium**	Secondary disclosure; lateral movement via synced data	Unmanaged devices; unrestricted clipboard/browser sync	Managed endpoints/VDI; disable/scope clipboard & sync; cache hygiene	Clipboard Data Attacks and Detection via Remote Desktop Protocol (Mohamed et al., 2023)

Taken together, the pathways in Table 1 demonstrate that "employee-driven leakage" is not a single action but a heterogeneous ecology of work practices spanning prompts, file uploads, plug-ins, API relays, sharing links, and residual endpoint artefacts. The conventional risk narrative—an engineer copy-pasting proprietary text into a

public chatbot—represents only the most visible portion of the surface. Once screenshots, bulk document uploads, wrapper-mediated calls and auto-ingesting assistants are considered, a substantial fraction of exposure moves into channels where payloads are either invisible to existing controls (e.g. text-only DLP confronted with OCR) or fragmented across multiple vendor systems and log stores. Blocking a well-known LLM domain therefore addresses the least subtle vector while leaving higher-volume leakage paths unaffected.

A clear structural pattern is the systematic misalignment between observability and impact. Direct prompt pasting is high-frequency and high-impact but still offers a single egress point that can be monitored or rate-limited (Ray, 2023; Williams et al., 2025). By contrast, plug-ins, multi-vendor wrappers, and internal API relays diffuse responsibility and auditability, converting transient request payloads into durable, sensitive logs (Rathod et al., 2025). Email and document assistants expand the threat from point leakage events to corpus-wide ingestion, ingesting entire drives or mailboxes under opaque retention (Baek et al., 2025; R. Yang et al., 2025). Shared links and permissive transcript exports quietly widen the audience for technical documentation long after any deliberate interaction with an LLM has ended (Alzamil et al., 2025). Endpoint artefacts such as clipboard history, browser caches and synchronised profiles extend the temporal window of exposure, surfacing later in support, legal hold, or on compromised devices (Chivers & Hargreaves, 2011; Hur et al., 2023; Mendoza et al., 2015; Oh et al., 2011; Okolica & Peterson, 2011).

Recurrent control failures reinforce these pathways. Organisations continue to rely on static prohibitions and awareness messaging, while genuine friction is absent at the point where a user uploads, pastes, or shares proprietary material (Taeihagh, 2025). Technical safeguards remain concentrated around narrow choke points (e.g. DNS blocks) rather than at the high-leverage layers highlighted in the table: OCR-aware inspection, scoped OAuth permissions, curated plug-in catalogues, controlled secret handling, and link-expiry enforcement (Bhushan, 2025; Challappa et al., 2025; K. Chen et al., 2025). Most of these mitigations already exist in enterprise DLP suites, identity platforms, and endpoint management systems, but are rarely configured with LLM-mediated documentation flows in mind.

In sum, Table 1 operationalises what this integrative review refers to as employee-driven leakage of technical documentation into general-purpose LLMs. Each pathway is initiated or sustained by routine employee actions rather than adversarial compromise, and the information at risk consists primarily of specifications, SOPs, design notes, logs, and other engineering artefacts that constitute organisational intellectual property. Crucially, these artefacts are being routed—directly via prompts and uploads, or indirectly via plug-ins, API relays, and assistant ingestion—into general-purpose LLM ecosystems whose retention, secondary use, and vendor chains are only partially observable. The remainder of this integrative review examines how these leakage patterns arise from misaligned incentives, workflow convenience, permissive defaults, and governance assumptions that conceptualise “LLM risk” too narrowly.

4. System model and cross-cutting risks

The leakage pathways identified in Section 3 can be interpreted within a compact system model that captures (i) how technical documentation moves through its lifecycle, (ii) where trust boundaries are crossed, and (iii) which risk dimensions consequently arise. This model provides the structural basis for the evaluation framework in Section 6.

4.1. Documentation lifecycle, information states, and trust boundaries

For governance purposes, technical documentation can be treated as moving through five functional phases: (1) creation and revision, (2) storage and indexing, (3) access and transformation, (4) integration into official artefacts, and (5) release or archival (Hullavarad et al., 2015; Mokhtar & Yusof, 2015; Salminen et al., 2014; Sovrano et al., 2025). What changes with general-purpose LLM adoption is not the existence of these phases but the speed and opacity with which documentation can traverse them, often without the metadata, access controls, and audit traces that traditionally signal a boundary crossing (Karras et al., 2025; Taeihagh, 2025). In LLM-integrated applications, “documentation” routinely becomes both data and instructional substrate; prompt injection work has shown that untrusted content can be interpreted as control text, collapsing a boundary that conventional document workflows assume is stable (Greshake et al., 2023; Liu et al., 2024).

Across these phases, four transition points repeatedly cross trust boundaries in real organisations: (i) copy–paste of excerpts into chat interfaces, (ii) file or screenshot uploads (including image-to-text extraction), (iii) connector or plug-in invocation that grants broad repository access, and (iv) paste-back of model outputs into internal

repositories, tickets, or version control. The first three transitions move artefacts outward (from enterprise governance into external service stacks or multi-vendor toolchains); the fourth moves artefacts inward again, but now as LLM-mediated derivatives whose provenance and transformation chain are difficult to evidence (Taeihagh, 2025; B. Yang et al., 2025). Tool-using "agent" architectures further widen the boundary surface: distinct stages (system prompt, user prompt handling, memory retrieval, and tool usage) create multiple injection and poisoning opportunities, including scenarios where a poisoned memory store or tool response steers downstream actions (H. Zhang et al., 2025).

To make these transitions tractable, artefacts can be represented in four information states: raw proprietary (full internal documents), sensitive-derived (excerpts, logs, contextual summaries that still reveal operational or design intent), redacted (structure retained, sensitive values removed), and public (already disclosed) (Feretzakis, Papaspyridis, et al., 2024; Feretzakis, Verykios, et al., 2024). The central governance problem is that most real work pressure pushes users to operate in the raw-proprietary and sensitive-derived states, exactly where LLM-mediated transformations are least observable and most consequential (Pahune et al., 2025; Waters-Lynch et al., 2025). Moreover, trust boundaries are not purely "inside vs. outside": LLM-integrated applications can bridge to internal systems (e.g., databases) in ways that reintroduce classic injection risks through natural-language interfaces (Pedro et al., 2025). Model-side leakage, extraction, and tool-mediated spillover risks are defined and evidenced in Section5.3; here we treat them only as a downstream risk class that becomes relevant once artefacts enter LLM interactions.

4.2. Provider data handling and residual retention

Once documentation crosses an enterprise boundary, it is processed within a multi-layer service stack that typically includes (i) an inference layer that receives prompts and uploaded artefacts, (ii) safety/abuse-monitoring and operational logging pipelines, and (iii) optional analytics or product-improvement pathways whose activation depends on plan, settings, and contract terms (Pahune et al., 2025; Shvetsova et al., 2025). In practice, "no-training" or "do not use for model improvement" controls should be interpreted narrowly: they may constrain whether inputs/outputs are used to update model parameters, but they do not necessarily eliminate short-term retention for abuse monitoring, incident response, debugging, or compliance workflows (Malki et al., 2025b; A. Zhang, 2025). For example, OpenAI's API documentation notes that abuse-monitoring logs may include prompts and responses and are retained for up to 30 days by default, with stricter options (e.g., zero data retention) available only under additional requirements and prior approval (*Data Controls in the OpenAI Platform*, n.d.). Similarly, enterprise-oriented offerings may provide administrator-configurable retention and deletion behaviour, but retention can still be affected by legal obligations and operational constraints (*Enterprise Privacy at OpenAI*, n.d.). In parallel, enterprise copilots can introduce their own retention and compliance storage paths (e.g., mailbox-based retention and eDiscovery handling), which means that "what users see" in a chat UI is not a reliable indicator of what is retained for governance purposes (Ishrak Alim, 2025; Sai et al., 2024). Finally, when LLM use is mediated by plug-ins, connectors, or third-party assistants, the chain of processing can expand to additional controllers/processors and subprocessors, increasing the difficulty of end-to-end accountability and DPIA-style mapping of data flows (Das et al., 2025). These handling uncertainties do not imply adversarial intent; rather, they constitute a structural opacity that, combined with the leakage pathways under the Section 3, creates non-trivial residual exposure risk that must be managed explicitly (through minimisation, segmentation, retention controls, and auditable governance).

4.3. Cross-cutting risk dimensions

The boundary crossings in Section4.1–4.2 and the pathways in Section3 generate four recurring risk dimensions: (i) confidentiality/competitive exposure, (ii) compliance and cross-border transfer, (iii) model-side effects (extraction, spillover, contamination), and (iv) incentives and governance gaps. These dimensions are analytically distinct but operationally coupled in multi-component LLM service stacks (UI, middleware, connectors, tool calls, and logging). Section5 defines each dimension and specifies the evaluation questions used in Section6.

5. Risk dimensions

This section defines the four risk dimensions precisely enough to be operationalised in Section6 for evaluating real documentation workflows. Each dimension is specified in terms of mechanisms, observable indicators, and governance-relevant control points within LLM-mediated service stacks.

5.1. Confidentiality and competitive exposure

Confidentiality risk in documentation-heavy work is rarely limited to "obvious secrets." LLM-assisted handling can create a cumulative "mosaic risk," in which seemingly low-sensitivity fragments—parameter defaults, troubleshooting sequences, tolerance bands, supplier identifiers, or sequencing logic—become sensitive when aggregated across repeated interactions, sessions, recipients, and time (Agarwal et al., 2024; Staab et al., 2024; Wang et al., 2025). The governance-relevant point is not deterministic trade-secret disclosure from any single snippet, but the progressive erosion of compartmentalisation and evidentiary control: repeated transfers can reconstruct design intent and tacit know-how while making it harder to demonstrate that reasonable secrecy measures were maintained (Nealey et al., 2015; Ozcan et al., 2025). In practical terms, weakly governed externalisation into third-party LLM workflows can undermine the defensibility of secrecy claims, especially where organisations cannot later evidence what was disclosed, under what contractual terms, and with what retention constraints (Aplin et al., 2023; Ozcan et al., 2025).

A second mechanism is reverse exposure via "paste-back." When employees integrate model outputs into internal artefacts (tickets, runbooks, manuals, deployment guides), organisations may import third-party or previously exposed proprietary phrasing, distinctive parameter values, or unattributed code fragments into governed repositories, complicating provenance, access control, and downstream sharing (Feretzakis et al., 2025; Perry et al., 2023). Even if such events are individually low probability, they can be hard to detect, trace, and quarantine at scale—particularly when productivity pressure normalises "copy–modify–ship" behaviours (Brynjolfsson et al., 2025). In multi-vendor plug-in ecosystems, the confidentiality boundary is further weakened by indirect prompt injection and tool calls: untrusted external content can steer what is retrieved or summarised and thereby increase the chance of disclosing internal context available via chat history, attachments, or connected tools (T. Chen et al., 2025; Zhan et al., 2024).

5.2. Compliance and cross-border transfer

Compliance risk in LLM-mediated documentation workflows is driven less by intent than by opacity and distributed processing. Technical documentation, operational logs, and incident records frequently embed personal data (e.g., names, emails, chat excerpts), access tokens in traces, device or account identifiers, and sometimes regulated or contractually restricted technical information. Once such materials cross an enterprise boundary into an LLM service stack, organisations can struggle to demonstrate controller–processor discipline because responsibilities distribute across controllers, processors, and sub-processors, while content may be replicated into telemetry, abuse monitoring, and debugging workflows that are only partially observable to the organisation (Feretzakis et al., 2025; Kramcsák, 2023). GDPR-oriented analyses argue that this stack complexity creates practical obstacles for purpose limitation, minimisation, retention limitation, and the execution—and evidencing—of access, rectification, erasure, and broader accountability obligations (Feretzakis et al., 2025; Kuru, 2024). Even where "no-training" is contractually specified, the operational compliance question remains whether the organisation can reliably map where the data travelled, which entities processed it, which jurisdictions stored it, and how deletion or DSAR-like obligations would be executed across all replicas (Feretzakis et al., 2025; Kramcsák, 2023).

Cross-border transfer is therefore best treated as a sub-problem of the same opacity: data residency and onward-transfer constraints may be difficult to evidence when content is routed through multiple subprocessors or persists in operational logs across regions, regardless of whether inputs are excluded from model-parameter updates. The governance gap is thus not only "what the model learns," but "where the content travels, where it rests, and how deletion claims can be substantiated." Legal scholarship further notes that the GDPR's consent model is often a poor fit for many AI contexts, reinforcing the need to minimise personal data in prompts and to treat multi-processor and cross-border flows as first-class risk factors rather than documentation footnotes (Kramcsák, 2023). For documentation-rich workflows, this pushes compliance away from abstract policy statements and toward concrete, auditable workflow design—classification rules, pre-submission redaction, approved routing, and verifiable retention controls (Feretzakis et al., 2025).

5.3. Model-side effects (spillover, extraction, contamination)

Model-side effects concern what can happen inside, or because of, the model and its surrounding agent/tool environment once proprietary or regulated artefacts enter LLM interactions. The strongest empirical basis is not general speculation about "models will leak everything," but demonstrated leakage mechanisms under realistic

threat models, including privacy leakage and extractable memorisation (Rathod et al., 2025). Empirical work shows that alignment does not eliminate extraction risk: under some conditions, adversaries can elicit memorised training samples from aligned, production-grade models, which makes a residual (non-zero) leakage probability a defensible governance assumption even when providers deploy safety layers (Nasr et al., 2025). Complementary evidence indicates that language models can leak personally identifiable information and that such leakage can be probed and measured; large-scale analyses characterise PII exposure modes, and tools such as ProPILE operationalise tests for whether PII is retrievable from LLM-based services (S. Kim et al., 2023; Lukas et al., 2023). Beyond memorisation, privacy can also be violated through inference: models may infer sensitive attributes from text at inference time, extending the risk surface from "did the model memorise it?" to "can the model deduce it?" (Lukas et al., 2023; Staab et al., 2024). The implication for documentation leakage is bounded but concrete: the relevant risk is not guaranteed disclosure, but an irreducible residual chance that portions of submitted content become recoverable or reconstructible in ways that are difficult to attest, reverse, or independently audit (Nasr et al., 2025). In the system model of Section4, these effects are downstream of outward boundary crossings and must therefore be assessed together with retention/logging and tool-access decisions.

A second mechanism is tool-mediated spillover in LLM-integrated applications and agents. Prompt-injection research shows that when LLMs are embedded in pipelines that ingest untrusted external content, attackers can manipulate model behaviour to exfiltrate data or trigger unintended actions by exploiting the blurred boundary between "instructions" and "data" (Greshake et al., 2023; Liu et al., 2024). In enterprise documentation settings, the practical concern is not deterministic cross-customer spillover, but limited attestability: organisations may be unable to prove that proprietary fragments did not influence future outputs, audits, or emergent tool-mediated behaviours—especially in systems vulnerable to indirect prompt injection and retrieval/tool contamination (Zhan et al., 2024). Agent-centric benchmarking further indicates that tool access and memory mechanisms expand the attack surface and require evaluation of attacks and defences at the agent layer, not only at the base model (H. Zhang et al., 2025). The governance consequence is structural: if organisations cannot bound which inputs are trusted, what tools can be invoked, and what logs persist, then "safe use" claims cannot be substantiated without explicit constraints, monitoring, and audit artefacts—criteria operationalised in Section6.

5.4. Incentives and governance gaps

The persistence of the high-impact leakage pathways in documentation-heavy work is best explained by an incentives-and-capability mismatch. Activities such as debugging, configuration, incident response, and standards interpretation carry high time pressure and cognitive load, while general-purpose LLM tools can deliver measurable productivity and speed-of-resolution gains in real workplace deployments—often disproportionately benefiting less experienced staff (Brynjolfsson et al., 2025). This creates strong local incentives to externalise real work artefacts into LLM tools even when formal governance is incomplete. Organisational scholarship on covert or "shadow" generative-AI use similarly finds that, where sanctioned alternatives are slow, ambiguous, absent, or cumbersome, employees predictably route around policy using unapproved tools, personal accounts, or plug-ins because the perceived short-term benefit dominates abstract policy risk (Waters-Lynch et al., 2025). In that setting, policy-only prohibitions are not credible: they do not remove demand for rapid interpretation of error traces, runbooks, and documentation; they mainly displace the behaviour into less observable channels (Waters-Lynch et al., 2025).

The practical implication is that leakage risk is not only a technical control problem but a workflow-design problem. Organisations must either provide sanctioned, auditable tooling that matches the productivity utility of consumer-grade LLMs, or accept that unsanctioned pathways will remain active and will dominate precisely in high-pressure contexts (incidents, outages, escalations) where documentation is most sensitive (Brynjolfsson et al., 2025; Waters-Lynch et al., 2025). For this reason, the evaluation framework in Section6 should treat incentives, usability, and governance capacity as first-class risk controls, alongside technical restrictions and monitoring, rather than relying on awareness training as the primary mitigation.

6. Evaluation framework for organisations

This section operationalises RQ4 by translating the pathway typology and cross-cutting risk dimensions into a reproducible governance rubric—qualitative scoring, rule-based escalation to a minimal guardrail stack, and a small set of metrics that can be tracked over time. The purpose is not to "measure" leakage with spurious precision, but to make risk assumptions auditable under policy volatility, multi-vendor processing chains, and strong employee incentives to externalise cognitive work into general-purpose LLMs. The unit of analysis is the

transition point where documentation crosses a trust boundary (copy–paste, upload/OCR, connector invocation, and paste-back). Controls should therefore be evaluated primarily at these transitions, not only at coarse network egress. This emphasis is consistent with the literature showing that LLM-integrated workflows introduce additional attack surfaces—particularly indirect prompt injection, tool-chain manipulation, and memory poisoning—whose observability and responsibility attribution are structurally weak in multi-component stacks (Greshake et al., 2023; Liu et al., 2024; A. Zhang, 2025). Where organisations use retrieval-augmented generation (RAG) and tool-using agents, evaluation must treat external content, tool responses, and "memory" stores as adversarially influenceable inputs, not merely benign context (Greshake et al., 2023; H. Zhang et al., 2025).

6.1. Risk scoring

Risk is scored on severity and likelihood, adjusted by an observability modifier, and accompanied by a confidence tag. Severity captures the sensitivity and blast radius of the artefact class (including trade-secret core, safety-critical, or export-controlled categories); likelihood captures workflow friction and incentive alignment (low-friction, high-speed actions score highest); observability captures whether detection is prompt and reliable or plausibly absent (e.g., wrapper relays, local artefacts, opaque tool chains). The priority score is defined as:

$$P = (S \times L) + O, \text{ with } S \in \{1,2,3,4\}, L \in \{1,2,3,4\}, \text{ and } O \in \{-1,0,+1\}. \quad (1)$$

- **Severity** $S \in \{1,2,3,4\}$: sensitivity and potential blast radius (1: routine internal; 2: internal but non-critical; 3: commercially sensitive; 4: trade-secret cores, export-controlled, or safety-critical artefacts);
- **Likelihood** $L \in \{1,2,3,4\}$: pathway friction and incentives (1: architecturally unlikely; 2: possible with effort/exception; 3: common workflow, weak guardrails; 4: low-friction, fast local action);
- **Observability modifier** $O \in \{-1,0,+1\}$: subtract 1 if promptly visible (e.g., gateway inspection); add 1 if detection is improbable (e.g., wrapper relays, endpoint artefacts);
- **Confidence tag** $C \in \{\text{low, medium, high}\}$: evidence quality for the assigned S,L,O. When confidence is low, organisations should either (i) treat the score as provisional and prioritise evidence collection (logs, sampling, red-team tests), or (ii) apply the next-higher tier as a conservative default until confidence improves.

The central discipline is interpretive: P is a triage band, not a probability. For model-side concerns, organisations should explicitly distinguish (i) direct disclosure (employee sends proprietary content outward), (ii) indirect disclosure (content later appears through logs, tool chains, or sharing defaults), and (iii) model-mediated effects with bounded but non-zero plausibility (memorisation and extraction; privacy inference; prompt-driven exfiltration). The peer-reviewed literature does not justify deterministic leakage claims, but it does justify treating extraction and inference risks as "credible under some conditions," including for aligned or production systems and for PII-like sequences (Lukas et al., 2023; Staab et al., 2024; Nasr et al., 2025).

6.2. Minimal guardrail stack (rule-based escalation)

Tiering is a governance choice: it makes enforcement discussable and testable, and it reduces the common failure mode in which organisations maintain a prohibition while tolerating unmanaged shadow use because sanctioned workflows are slower or absent (Waters-Lynch et al., 2025). Tier 0 should be universal and low-friction: sanctioned endpoints and identity-bound access; least-privilege scopes for connectors; point-of-use classification and just-in-time prompts before paste/upload; and contractual commitments on retention, sub-processing, and training exclusion for customer inputs. If Tier 0 does not provide a usable compliant path for the dominant tasks (summarisation, troubleshooting, standards interpretation), it should be treated as ineffective by design and expected to increase unsanctioned tool use. Tier 1 adds inspection and minimisation where impact or invisibility is high: prompt/file gateways with OCR-aware inspection; metadata stripping; client-side redaction templates; curated plug-in catalogues with kill switches; and logging minimisation with targeted redaction. Tier 2 is reserved for the highest-consequence classes: architectural reduction via private or enterprise deployments with verifiable residency/retention, proxy-mediated key management, and periodic verification (including deletion tests and change-log review).

Escalation rules should remain simple and auditable rather than "optimised." A defensible baseline is: apply Tier 0 universally; add Tier 1 when (a) P is high, (b) severity and likelihood are both high, or (c) observability is poor; add Tier 2 whenever S = 4 (even if likelihood is moderate) or where regulated/export-controlled classes are plausibly in scope. In tool-using or RAG-enabled systems, Tier 1–2 decisions should weight prompt-injection

evidence more heavily because the literature demonstrates that "data-as-instructions" attacks can turn retrieved documents or tool outputs into control channels.

6.3. Escalation rules (auditable, not optimised):

- Apply Tier 0 universally.
- Add Tier 1 if $P \geq 9$ or and ($S \geq 3$ and $L \geq 3$) or O=+1.
- Add Tier 2 if $S = 4$ with $L \geq 2$, or whenever regulated/export-controlled data are in scope, irrespective of P.

These rules prioritise high-consequence, high-likelihood, low-visibility pathways and concentrate enforcement at transition points (before prompts/uploads and before paste-back).

6.4. Provider and architecture due diligence questions

Because provider practices and plugin ecosystems shift, due diligence should be stated as questions that can be contractually answered and periodically re-validated. Minimum questions include: (i) what inputs are retained, where, and for how long (prompts, uploads, safety traces, telemetry, tool outputs); (ii) what is excluded from model improvement and what is merely excluded from parameter updates; (iii) what sub-processors and tool vendors receive data under connectors; (iv) what audit evidence exists (tenant logs, deletion attestations, scoped keys, and control-plane enforcement); (v) what is the failure mode under prompt injection and tool compromise (output handling, tool permissions, memory poisoning) and (vi) what controls exist to detect and govern paste-back of model outputs into internal repositories (e.g., watermarking/provenance metadata, DLP on commits/tickets), and how are exceptions audited? The point is not to assume malice; it is to treat opacity itself as a risk amplifier, particularly for high-value documentation classes.

6.5. Metrics (definitions and formulas)

Metrics should be few, operationally collectible, and interpretable as trends rather than absolutes. Windows of observation (e.g., rolling 30 days) and inclusion criteria should be specified by the organisation. To make metrics comparable over time, the organisation should version the measurement specification (data sources, sampling, and what counts as an "interaction") and record any changes alongside the trend charts.

- Coverage of sanctioned egress (COV_{egress}):

$$Cov_{egress} = \frac{\text{request to allow-listed LLM endpoints}}{\text{all detected LLM requests}} \tag{2}$$

- Adoption ratio (AR) (sanctioned vs. unsanctioned tools):

$$AR = \frac{\text{session using sanctioned endpointss/tools}}{\text{session using sanctioned + unsanctioned tools}} \tag{3}$$

- Blocking effectiveness (BE) (policy-relevant events):

$$BE = \frac{\text{blocked outbound events}}{\text{blocked + allowed outbound events}} \tag{4}$$

- False-positive rate of inspection (FPR):

$$FPR = \frac{\text{non-sensitive items flagged}}{\text{all items flagged}} \tag{5}$$

- Drift lag (DL) (policy responsiveness to external change):

$$DL = t_{internalupdate} - t_{providerchange} \tag{6}$$

- Mean time to compliant alternative ($MTTC$):

$$MTTC = \frac{1}{N}\sum_{i=1}^{N}\left(t_{approved}^{(i)} - t_{request}^{(i)}\right) \quad (7)$$

Interpretation is comparative rather than absolute. Improvements in coverage (Cov_{egress}) and adoption ratio (*AR*) indicate consolidation onto governed channels; increases in blocking effectiveness (*BE*) coupled with stable or declining false-positive rates (*FPR*) suggest effective, non-disruptive inspection; reductions in drift lag (*DL*) and mean time to compliant alternative (*MTTC*) indicate responsiveness and usability of sanctioned paths. The leakage typology mapped in Table 1 should be maintained as a living register; when architecture, controls, or provider commitments change within a reporting window, update entries, note the change date, and recompute priority bands with the same rubric so that metric shifts can be interpreted against governance changes.

7. Discussion

This integrative review set out to map how routine employee interactions with general-purpose LLMs can create bidirectional flows between internal documentation systems and external model ecosystems, and to translate that mechanism map into proportionate, auditable governance. The synthesis supports four main interpretations that jointly answer the research questions.

Regarding RQ1, the review indicates that leakage risk concentrates not in exotic attacks but in routine boundary crossings embedded in everyday documentation work: copy–paste of snippets into chat interfaces, upload of files and screenshots (including OCR-mediated extraction), connector-mediated retrieval across repositories, and paste-back of model outputs into internal tickets, runbooks, and repositories (Table 1; Section 4). These transitions are high-frequency, low-friction, and often weakly observable, which makes them structurally more consequential than isolated "full-document exfiltration" narratives. The implication is that governance must target *transition points* as the unit of control rather than attempting to "secure documentation" in the abstract. This conclusion remains bounded by the review's design: the paper maps plausible and documented mechanisms, but it does not estimate prevalence by sector or vendor.

With respect to RQ2, across the mapped pathways, four risk dimensions recur and compound: (i) confidentiality and competitive exposure through cumulative disclosure of seemingly minor fragments, (ii) compliance and cross-border transfer challenges driven by distributed processing chains and evidencing requirements, (iii) model-side effects that justify treating leakage/inference as bounded-but-nonzero residual risk, and (iv) incentive-driven governance gaps that sustain shadow workflows when sanctioned alternatives are absent or slower (Section 5). The persistence of these dimensions follows from two structural features: compositional service stacks (interfaces, logging/telemetry, plug-ins, relays) and limited organisational observability of where artefacts travel and persist once they cross the boundary (Section 4). The implication is that the same control family can reduce one dimension while worsening another (e.g., logging for security can amplify retention exposure), so an explicit multi-dimensional framing is necessary for credible trade-off management. This synthesis is constrained by heterogeneity in legal regimes and service configurations; therefore, the risk dimensions should be treated as a portable taxonomy rather than a uniform compliance verdict.

Addressing RQ3, the evidence supports a mechanism-centred governance approach that prioritises auditable guardrails at the transition points rather than policy-only prohibitions or coarse network blocking (Sections 4–6). The proposed "minimal guardrail stack" is defensible because it maps directly onto the failure modes in Table 1: classification and just-in-time friction before paste/upload, OCR-aware inspection and redaction, least-privilege connector scoping with curated plug-in catalogues, and logging minimisation in internal relays, with escalation for high-sensitivity artefacts (Section 6). The implication is that organisations reduce risk most credibly by consolidating employees onto sanctioned channels and constraining connector/tool permissions, while maintaining usable workflows to prevent displacement into less observable shadow use. This remains conditional: control effectiveness depends on implementation quality (especially inspection accuracy and connector governance) and on whether sanctioned alternatives match operational tempo in high-pressure contexts.

In response to RQ4, the paper's contribution is an evaluation framework that converts the pathway map and risk taxonomy into auditable organisational decision-making: qualitative scoring (severity, likelihood, observability, confidence), rule-based escalation to tiered controls, and a small set of governance metrics that track coverage, friction, drift, and time-to-compliant-alternative rather than attempting to quantify leakage prevalence (Section 6). This responds directly to the evidentiary reality that incidents are under-reported and downstream attribution is weak; therefore, a tractable framework should focus on what organisations can measure and improve—sanctioned-channel consolidation, transition-point block rate, inspection false positives, drift lag, and mean time to a compliant workflow—while keeping uncertainty explicit (Section 6; Limitations). The implication is that

"risk reduction" is operationalised as reduced boundary crossings for high-sensitivity content and increased auditability of those that remain, not as a claim that leakage has been eliminated. The framework is bounded by log availability and classification fidelity; where these are absent, metrics must be treated as indicators or established through sampling audits.

Two broader implications follow for scholarship and practice. For research, the review highlights a measurement gap: incident reporting is sparse, attribution beyond initial disclosure is rarely demonstrable, and the most consequential channels (plug-in chains, OCR, endpoint residue, internal relay logging) are under-studied relative to prompt text alone. This suggests value in organisational field studies of governed vs. unguided adoption, and in technical evaluations of inspection/redaction effectiveness for multimodal inputs and tool-using agents (Greshake et al., 2023; Liu et al., 2024; Zhang et al., 2025). For practice, the main warning is that "no-training" commitments, while relevant, do not resolve governance: the risk surface is dominated by boundary crossings, distributed processing, and weak observability, so controls must be designed for traceability and minimisation across the full workflow chain rather than anchored in a single provider claim. The paper's contribution is therefore best understood as a compact, mechanism-centred roadmap for cautious adoption: it does not assert prevalence or inevitability of leakage, but it makes the pathways, risk dimensions, and evaluation assumptions explicit enough to support auditable organisational decisions.

8. Limitations of the study

This study adopts an integrative review with a mechanism-oriented synthesis rather than a systematic review or meta-analysis. That choice is methodologically defensible for an emergent and policy-volatile domain, but it imposes clear constraints on inference. First, coverage is not exhaustive and selection is purposive: the corpus intentionally spans heterogeneous evidence types (peer-reviewed studies, preprints, provider documentation, regulatory guidance, and structured incident reporting), prioritising mechanism mapping over prevalence estimation. As a result, the paper should be read as a *structured plausibility argument* about how leakage can occur, not as an epidemiology of how often it occurs in any sector or region.

Second, time variance is a fundamental limitation. Provider terms, retention defaults, product architectures, and connector ecosystems change rapidly, and the same "LLM service" label can hide materially different handling depending on plan, configuration, geography, and integration choices. Accordingly, statements about provider handling are best interpreted as time-stamped observations, not stable properties of the ecosystem. This limits the durability of any provider-specific interpretation and implies that organisational governance must treat "drift" as expected rather than exceptional.

Third, incident evidence is structurally incomplete. Data exposure events are under-reported, frequently handled under confidentiality constraints, and often lack technical detail sufficient for attribution beyond the initial disclosure. Even when an incident is documented, chain-of-custody across plug-ins, wrappers, internal relays, and endpoint artefacts is rarely reconstructable. Consequently, the review cannot support strong causal claims about downstream reuse, cross-organisational propagation, or long-run model-mediated diffusion; it can only argue that certain pathways and residual risks remain plausible under bounded conditions.

Fourth, the qualitative risk scoring is inherently context-dependent. Severity, likelihood, and observability are assessed on ordinal scales as a transparent prioritisation rubric, but these judgements depend on sector, documentation criticality, workforce incentives, endpoint hygiene, and the organisation's actual tooling stack. The same pathway can shift bands when, for example, a firm moves from ad hoc public chat use to a sanctioned enterprise endpoint with constrained connectors and inspection, or conversely when tool-using agents and broad OAuth scopes are introduced. The framework is therefore not a universal "risk calculator"; it is a reproducible way to make assumptions explicit and to keep prioritisation consistent within a given organisation over time.

Fifth, generalisability across jurisdictions and organisational maturities is limited. The compliance dimension is shaped by local legal regimes (e.g., data transfer constraints, sectoral regulations, export-control rules), and the feasibility of controls is shaped by baseline governance capacity (identity, DLP, endpoint management, procurement leverage). The paper's control recommendations therefore represent a minimal, mechanism-centred guardrail stack that is intended to be contractable and auditable in principle, but it does not claim that all organisations can implement all tiers without material resourcing and change management.

Finally, the paper does not empirically validate control effectiveness. Proposed mitigations (e.g., OCR-aware inspection, connector scoping, curated plug-in catalogues, logging minimisation, and tiered architectural reduction) are grounded in known control families and the mapped pathways, but the review does not provide

experimental measurements of false positives/negatives, usability impacts, or leakage reduction in operational deployments. That validation is a concrete direction for future work: controlled studies and field evaluations comparing governed vs. unguided adoption, especially under high-pressure operational contexts where incentives and sensitivity peak.

9. Conclusions

This paper maps the mechanisms through which the routine use of general-purpose LLMs by employees creates bidirectional data flows between internal documentation systems and external model ecosystems. The evidence suggests that the primary risks reside not in isolated incidents but in recurring transition points—such as copy–paste, file uploads, and connector invocation—integrated into daily workflows. The analysis confirms that four core risk dimensions (confidentiality, compliance, model-side effects, and governance gaps) require a transition from reactive prohibitions to proactive, workflow-based governance. The proposed "minimal guardrail stack" and evaluation framework offer organisations a verifiable method for risk triage and cautious adoption. Instead of claiming definitive leakage rates, this work provides a roadmap for explicit risk-based decision-making in a volatile technological landscape.

Author Contributions

Conceptualisation, L.L. and O.Z.; methodology, L.L.; validation, O.M., D.Ž.; investigation, L.L., O.Z., O.M., D.Ž.; data curation, O.M. and D.Ž; writing—original draft preparation, L.L.; writing—review and editing, O.Z., O.M; supervision, D.Ž. All authors have read and agreed to the published version of the manuscript.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflicts of Interest

The authors declare no conflict of interest.

Literature

1. Acar, G., Englehardt, S., & Narayanan, A. (2020). No boundaries: Data exfiltration by third parties embedded on web pages. *Proceedings on Privacy Enhancing Technologies*. https://petsymposium.org/popets/2020/popets-2020-0070.php
2. Agarwal, D., Fabbri, A., Risher, B., Laban, P., Joty, S., & Wu, C.-S. (2024). Prompt Leakage effect and mitigation strategies for multi-turn LLM Applications. In F. Dernoncourt, D. Preoțiuc-Pietro, & A. Shimorina (Eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track* (pp. 1255–1275). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.emnlp-industry.94
3. Alzamil, L. M., Alhasani, A. M., Alshehri, S., Alzamil, L. M., Alhasani, A. M., & Alshehri, S. (2025). Privacy Concerns in ChatGPT Data Collection and Its Impact on Individuals. *Future Internet*, *17*(11). https://doi.org/10.3390/fi17110511
4. Aplin, T., Radauer, A., Bader, M. A., & Searle, N. (2023). The Role of EU Trade Secrets Law in the Data Economy: An Empirical Analysis. *IIC - International Review of Intellectual Property and Competition Law*, *54*(6), 826–858. https://doi.org/10.1007/s40319-023-01325-8
5. Baek, S. J., Lee, H. J., Baek, S. J., & Lee, H. J. (2025). Unravelling the Effects of Privacy Policies on Information Disclosure: Insights from E-Commerce Consumer Behavior. *Journal of Theoretical and Applied Electronic Commerce Research*, *20*(1). https://doi.org/10.3390/jtaer20010049
6. Bhardwaj, A., & Goundar, S. (2017). Security challenges for cloud-based email infrastructure. *Network Security*, *2017*(11), 8–15. https://doi.org/10.1016/S1353-4858(17)30094-6
7. Bhushan, B. (2025). An Explainable Zero Trust Identity Framework for LLMs, AI Agents, and Agentic AI Systems. *International Journal of Computer Applications*, *187*(46), 42–52.
8. Brynjolfsson, E., Li, D., & Raymond, L. (2025). Generative AI at Work*. *The Quarterly Journal of Economics*, *140*(2), 889–942. https://doi.org/10.1093/qje/qjae044

9. Challappa, L., Zhang, Z., & Garg, R. (2025). Domain anchorage in LLMs: Lexicon profiling and unintended information leakage. *Data & Policy*, *7*, e73. https://doi.org/10.1017/dap.2025.10041
10. Chen, K., Zhou, X., Lin, Y., Feng, S., Shen, L., & Wu, P. (2025). A survey on privacy risks and protection in large language models. *Journal of King Saud University Computer and Information Sciences*, *37*(7), 163. https://doi.org/10.1007/s44443-025-00177-1
11. Chen, T., Li, P., Zhou, K., Chen, T., & Wei, H. (2025). Unveiling Privacy Risks in Multi-modal Large Language Models: Task-specific Vulnerabilities and Mitigation Challenges. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Findings of the Association for Computational Linguistics: ACL 2025* (pp. 4573–4586). Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.findings-acl.237
12. Chivers, H., & Hargreaves, C. (2011). Forensic data recovery from the Windows Search Database. *Digital Investigation*, *7*(3), 114–126. https://doi.org/10.1016/j.diin.2011.01.001
13. Clusmann, J., Ferber, D., Wiest, I. C., Schneider, C. V., Brinker, T. J., Foersch, S., Truhn, D., & Kather, J. N. (2025). Prompt injection attacks on vision language models in oncology. *Nature Communications*, *16*(1), 1239. https://doi.org/10.1038/s41467-024-55631-x
14. Das, B. C., Amini, M. H., & Wu, Y. (2025). Security and Privacy Challenges of Large Language Models: A Survey. *ACM Comput. Surv.*, *57*(6), 152:1-152:39. https://doi.org/10.1145/3712001
15. *Data controls in the OpenAI platform*. (n.d.). Retrieved December 23, 2025, from https://platform.openai.com
16. Ebert, C. (2013). Improving engineering efficiency with PLM/ALM. *Software & Systems Modeling*, *12*(3), 443–449. https://doi.org/10.1007/s10270-013-0347-3
17. *Enterprise privacy at OpenAI*. (n.d.). Retrieved December 23, 2025, from https://openai.com/enterprise-privacy/
18. Feretzakis, G., Papaspyridis, K., Gkoulalas-Divanis, A., Verykios, V. S., Feretzakis, G., Papaspyridis, K., Gkoulalas-Divanis, A., & Verykios, V. S. (2024). Privacy-Preserving Techniques in Generative AI and Large Language Models: A Narrative Review. *Information*, *15*(11). https://doi.org/10.3390/info15110697
19. Feretzakis, G., Vagena, E., Kalodanis, K., Peristera, P., Kalles, D., Anastasiou, A., Feretzakis, G., Vagena, E., Kalodanis, K., Peristera, P., Kalles, D., & Anastasiou, A. (2025). GDPR and Large Language Models: Technical and Legal Obstacles. *Future Internet*, *17*(4). https://doi.org/10.3390/fi17040151
20. Feretzakis, G., Verykios, V. S., Feretzakis, G., & Verykios, V. S. (2024). Trustworthy AI: Securing Sensitive Data in Large Language Models. *AI*, *5*(4), 2773–2800. https://doi.org/10.3390/ai5040134
21. Ferrag, M. A., Tihanyi, N., Hamouda, D., Maglaras, L., Lakas, A., & Debbah, M. (2025). From prompt injections to protocol exploits: Threats in LLM-powered AI agents workflows. *ICT Express*. https://doi.org/10.1016/j.icte.2025.12.001
22. Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. (2023). Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*, 79–90. https://doi.org/10.1145/3605764.3623985
23. Hullavarad, S., O'Hare, R., & Roy, A. K. (2015). Enterprise Content Management solutions—Roadmap strategy and implementation challenges. *International Journal of Information Management*, *35*(2), 260–265. https://doi.org/10.1016/j.ijinfomgt.2014.12.008
24. Hur, U., Kang, S., Kim, G., & Kim, J. (2023). A study on cloud data access through browser credential migration in Windows environment. *Forensic Science International: Digital Investigation*, *45*, 301568. https://doi.org/10.1016/j.fsidi.2023.301568
25. *Incident 768: ChatGPT Implicated in Samsung Data Leak of Source Code and Meeting Notes*. (2023). https://incidentdatabase.ai/cite/768/
26. *Incident 1186: Reported Public Exposure of Over 100,000 LLM Conversations via Share Links Indexed by Search Engines and Archived*. (2025). https://incidentdatabase.ai/cite/1186/
27. Ishrak Alim, T. F. M., Takib Md Masudul Hasan Prodhan ,. Md Lahaduzzaman Lahad. (2025). *The Insider Risk of Artificial Intelligence in Financial Systems through the Lens of Large Language Models*. https://doi.org/10.5281/zenodo.16910637
28. Karras, A., Theodorakopoulos, L., Karras, C., Krimpas, G. A., Giannaros, A., Bakalis, C.-P., Karras, A., Theodorakopoulos, L., Karras, C., Krimpas, G. A., Giannaros, A., & Bakalis, C.-P. (2025). LLM-Driven Big Data Management Across Digital Governance, Marketing, and Accounting: A Spark-Orchestrated Framework. *Algorithms*, *18*(12). https://doi.org/10.3390/a18120791

29. Kim, S., Yun, S., Lee, H., Gubri, M., Yoon, S., & Oh, S. J. (2023). ProPILE: Probing Privacy Leakage in Large Language Models. *In Large Language Models. NeurIPS 2023 Proceedings.*
30. Kim, Y., Yoon, H.-J., & Lee, M.-H. (2015). *Stealthy Information Leakage from Android Smartphone through Screenshot and OCR*. 784–787. https://doi.org/10.2991/cmfe-15.2015.184
31. Kramcsák, P. T. (2023). Can legitimate interest be an appropriate lawful basis for processing Artificial Intelligence training datasets? *Computer Law & Security Review*, *48*, 105765. https://doi.org/10.1016/j.clsr.2022.105765
32. Kuru, T. (2024). Lawfulness of the mass processing of publicly accessible online data to train large language models. *International Data Privacy Law*, *14*(4), 326–351. https://doi.org/10.1093/idpl/ipae013
33. Liu, Y., Geng, R., Jia, J., & Gong, N. Z. (2024). *Formalizing and Benchmarking Prompt Injection Attacks and Defenses*.
34. Lukas, N., Salem, A., Sim, R., Tople, S., Wutschitz, L., & Zanella-Béguelin, S. (2023). *Analyzing Leakage of Personally Identifiable Information in Language Models* (No. arXiv:2302.00539). arXiv. https://doi.org/10.48550/arXiv.2302.00539
35. Malki, L. M., Polamarasetty, A., Hatamian, M., Warner, M., & Costanza, E. (2025a). Hoovered up as a data point: Exploring Privacy Behaviours, Awareness, and Concerns Among UK Users of LLM-based Conversational Agents. *Proceedings on Privacy Enhancing Technologies*. https://petsymposium.org/popets/2025/popets-2025-0160.php
36. Malki, L. M., Polamarasetty, A., Hatamian, M., Warner, M., & Costanza, E. (2025b). Hoovered up as a data point: Exploring Privacy Behaviours, Awareness, and Concerns Among UK Users of LLM-based Conversational Agents. *Proceedings on Privacy Enhancing Technologies*. https://petsymposium.org/popets/2025/popets-2025-0160.php
37. Mendoza, A., Kumar, A., Midcap, D., Cho, H., & Varol, C. (2015). BrowStEx: A tool to aggregate browser storage artifacts for forensic analysis. *Digital Investigation*, *14*, 63–75. https://doi.org/10.1016/j.diin.2015.08.001
38. Mohamed, K. F., AbdelBaki, N., & Shosha, A. (2023). Clipboard Data Attacks and Detection via Remote Desktop Protocol. *2023 5th Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, 98–102. https://doi.org/10.1109/NILES59815.2023.10296672
39. Mokhtar, U. A., & Yusof, Z. M. (2015). The requirement for developing functional records classification. *International Journal of Information Management*, *35*(4), 403–407. https://doi.org/10.1016/j.ijinfomgt.2015.04.002
40. Mousavi, Z., Islam, C., Babar, M. A., Abuadbba, A., & Moore, K. (2025). Detecting Misuse of Security APIs: A Systematic Review. *ACM Comput. Surv.*, *57*(12), 303:1-303:39. https://doi.org/10.1145/3735968
41. Nasr, M., Rando, J., Carlini, N., Hayase, J., Jagielski, M., Cooper, A. F., Ippolito, D., Choquette-Choo, C. A., Tramer, F., & Lee, K. (2025). *SCALABLE EXTRACTION OF TRAINING DATA FROM ALIGNED, PRODUCTION LANGUAGE MODELS*.
42. Nealey, T., Daignault, R. M., & Cai, Y. (2015). Trade Secrets in Life Science and Pharmaceutical Companies. *Cold Spring Harbor Perspectives in Medicine*, *5*(4), a020982–a020982. https://doi.org/10.1101/cshperspect.a020982
43. Oh, J., Lee, S., & Lee, S. (2011). Advanced evidence collection and analysis of web browser activity. *Digital Investigation*, *8*, S62–S70. https://doi.org/10.1016/j.diin.2011.05.008
44. Okolica, J., & Peterson, G. L. (2011). Extracting the windows clipboard from physical memory. *Digital Investigation*, *8*, S118–S124. https://doi.org/10.1016/j.diin.2011.05.014
45. Ozcan, O., Pickernell, D., & Bacon, E. (2025). Identifying trade secrets: Strategic process and challenges in the UK. *Technology Analysis & Strategic Management*, *0*(0), 1–18. https://doi.org/10.1080/09537325.2025.2489155
46. Pahune, S., Akhtar, Z., Pahune, S., & Akhtar, Z. (2025). Transitioning from MLOps to LLMOps: Navigating the Unique Challenges of Large Language Models. *Information*, *16*(2). https://doi.org/10.3390/info16020087
47. Pedro, R., Coimbra, M. E., Castro, D., Carreira, P., & Santos, N. (2025). Prompt-to-SQL Injections in LLM-Integrated Web Applications: Risks and Defenses. *In Proceedings of the IEEE/ACM International Conference on Software Engineering (ICSE 2025)*.
48. Perry, N., Srivastava, M., Kumar, D., & Boneh, D. (2023). Do Users Write More Insecure Code with AI Assistants? *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2785–2799. https://doi.org/10.1145/3576915.3623157

49. Prinz, K. D. (2025). Managing the legal risks of artificial intelligence on intellectual property and confidential information. *Consulting Psychology Journal*, *77*(2), 169–179. https://doi.org/10.1037/cpb0000287
50. Rathod, V., Nabavirazavi, S., Zad, S., & Iyengar, S. S. (2025). Privacy and Security Challenges in Large Language Models. *2025 IEEE 15th Annual Computing and Communication Workshop and Conference (CCWC)*, 00746–00752. https://doi.org/10.1109/CCWC62904.2025.10903912
51. Ray, P. P. (2023). ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*, *3*, 121–154. https://doi.org/10.1016/j.iotcps.2023.04.003
52. Sai, S., Yashvardhan, U., Chamola, V., & Sikdar, B. (2024). Generative AI for Cyber Security: Analyzing the Potential of ChatGPT, DALL-E, and Other Models for Enhancing the Security Space. *IEEE Access*, *12*, 53497–53516. https://doi.org/10.1109/ACCESS.2024.3385107
53. Sajjadi Mohammadabadi, S. M., Kara, B. C., Eyupoglu, C., Uzay, C., Tosun, M. S., & Karakuş, O. (2025). A Survey of Large Language Models: Evolution, Architectures, Adaptation, Benchmarking, Applications, Challenges, and Societal Implications. *Electronics*, *14*(18), 3580. https://doi.org/10.3390/electronics14183580
54. Saka, A., Taiwo, R., Saka, N., Salami, B. A., Ajayi, S., Akande, K., & Kazemi, H. (2024). GPT models in construction industry: Opportunities, limitations, and a use case validation. *Developments in the Built Environment*, *17*, 100300. https://doi.org/10.1016/j.dibe.2023.100300
55. Salminen, A., Jauhiainen, E., & Nurmeksela, R. (2014). A life cycle model of XML documents. *Journal of the Association for Information Science and Technology*, *65*(12), 2564–2580. https://doi.org/10.1002/asi.23148
56. Shvetsova, O., Katalshov, D., Lee, S.-K., Shvetsova, O., Katalshov, D., & Lee, S.-K. (2025). Innovative Guardrails for Generative AI: Designing an Intelligent Filter for Safe and Responsible LLM Deployment. *Applied Sciences*, *15*(13). https://doi.org/10.3390/app15137298
57. Sovrano, F., Hine, E., Anzolut, S., & Bacchelli, A. (2025). Simplifying software compliance: AI technologies in drafting technical documentation for the AI Act. *Empirical Software Engineering*, *30*(4), 91. https://doi.org/10.1007/s10664-025-10645-x
58. Staab, R., Vero, M., Balunovic, M., & Vechev, M. (2024). *BEYOND MEMORIZATION: VIOLATING PRIVACY VIA INFERENCE WITH LARGE LANGUAGE MODELS*.
59. Standing, C., & Kiniti, S. (2011). How can organizations use wikis for innovation? *Technovation*, *31*(7), 287–295. https://doi.org/10.1016/j.technovation.2011.02.005
60. Starov, O., & Nikiforakis, N. (2017). Extended Tracking Powers: Measuring the Privacy Diffusion Enabled by Browser Extensions. *Proceedings of the 26th International Conference on World Wide Web*, 1481–1490. https://doi.org/10.1145/3038912.3052596
61. Taeihagh, A. (2025). Governance of Generative AI. *Policy and Society*, *44*(1), 1–22. https://doi.org/10.1093/polsoc/puaf001
62. Torraco, R. J. (2005). Writing Integrative Literature Reviews: Guidelines and Examples. *Human Resource Development Review*, *4*(3), 356–367. https://doi.org/10.1177/1534484305278283
63. Wan, L., Wang, K., Wang, H., & Bai, G. (2024). Is It Safe to Share Your Files? An Empirical Security Analysis of Google Workspace. *Proceedings of the ACM Web Conference 2024*, 1892–1901. https://doi.org/10.1145/3589334.3645697
64. Wang, Z., Liu, T., Liu, Y., Zio, E., & Guan, X. (2025). Data Inference: Data Security Threats in the AI Era. *Engineering*, *52*, 29–33. https://doi.org/10.1016/j.eng.2025.08.007
65. Waters-Lynch, J., Allen, D. W. E., Potts, J., & Berg, C. (2025). *Shadow User Innovation: Governing Covert Generative-AI Use for Dynamic-Capability Renewal* (SSRN Scholarly Paper No. 5281695). Social Science Research Network. https://doi.org/10.2139/ssrn.5281695
66. Whittemore, R., & Knafl, K. (2005). The integrative review: Updated methodology. *Journal of Advanced Nursing*, *52*(5), 546–553. https://doi.org/10.1111/j.1365-2648.2005.03621.x
67. Williams, A., Fox, G., Amon, M. J., Tanni, T. I., & Solihin, Y. (2025). The GenAI networked privacy problem at work- How privacy knowledge and perceptions predict Generative AI disclosure in professional contexts. *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, 1–9. https://doi.org/10.1145/3706599.3719923
68. Yang, B., Dang, J., Liu, H., & Jin, Z. (2025). Advancing LLM-Generated Code Reliability: A Hybrid Approach for Hallucination Detection. *IEEE Transactions on Software Engineering*, 1–17. https://doi.org/10.1109/TSE.2025.3640641

69. Yang, J., Jin, H., Tang, R., Han, X., Feng, Q., Jiang, H., Zhong, S., Yin, B., & Hu, X. (2024). Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond. *ACM Trans. Knowl. Discov. Data*, *18*(6), 160:1-160:32. https://doi.org/10.1145/3649506
70. Yang, R., Fu, M., Tantithamthavorn, C., Arora, C., Vandenhurk, L., & Chua, J. (2025). RAGVA: Engineering retrieval augmented generation-based virtual assistants in practice. *Journal of Systems and Software*, *226*, 112436. https://doi.org/10.1016/j.jss.2025.112436
71. Zhan, Q., Liang, Z., Ying, Z., & Kang, D. (2024). *InjecAgent: Benchmarking Indirect Prompt Injections in Tool-Integrated Large Language Model Agents* (No. arXiv:2403.02691). arXiv. https://doi.org/10.48550/arXiv.2403.02691
72. Zhang, A. (2025). Information Retrieval in the Age of Generative AI: A Mismatch That Matters. *Legal Reference Services Quarterly*, *44*(3), 297–306. https://doi.org/10.1080/0270319X.2025.2536920
73. Zhang, H., Huang, J., Mei, K., Yao, Y., Wang, Z., Zhan, C., Wang, H., & Zhang, Y. (2025). *AGENT SECURITY BENCH (ASB): FORMALIZING AND BENCHMARKING ATTACKS AND DEFENSES IN LLM-BASED AGENTS*.
74. Zolkifli, N. N., Ngah, A., & Deraman, A. (2018). Version Control System: A Review. *Procedia Computer Science*, *135*, 408–415. https://doi.org/10.1016/j.procs.2018.08.19