RAG and ruin: why your existing controls may miss AI poisoning attacks

BY ELEANOR BARLOW · JULY 28, 2026

Want to read the latest version? Visit the link below:
https://www.intigriti.com/blog/business-insights/why-your-existing-controls-may-miss-ai-poisoning-attacks

Key takeaways

- RAG systems expand the application's trust boundary by adding external, mutable content to the model context. If a threat actor can influence what gets indexed and retrieved, they can influence what the model says or does.
- In simple QA systems, that may mean misinformation or unsafe recommendations.
- In agentic systems with tools and permissions, it can become data leakage, unauthorized actions, or operational compromise.
- Organizations need controls around ingestion, provenance, retrieval, permissions, monitoring, and AI-specific testing.

Why RAG systems are useful

Retrieval-Augmented Generation (RAG) is an AI architecture that combines a retrieval system with a large language model (LLM). Instead of relying only on what the LLM learned during training, RAG first retrieves relevant information from an external knowledge source and then uses that information to generate a more accurate response.

At its core, RAG uses a question-answering system that does something deceptively simple but remarkably powerful: instead of relying solely on what an LLM already "knows", it goes out and fetches real information from documents and knowledge bases before forming its response. For organizations sitting on vast stores of internal data, this changes everything and is arguably one of the most important concepts to emerge from the AI boom of the early 2020s.

Workflow in its simplest form:

1. **User asks a question**: "What is our company's vacation policy?"
2. **Retrieve relevant documents**: The system searches a knowledge base for the most relevant passages.
3. **Augment the prompt**: The retrieved documents are added to the prompt sent to the LLM.
4. **Generate the answer**: The LLM produces a response based on both the user's question and the retrieved context.

Lewis *et al.* helped popularise the term in their 2020 paper, Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.

> **'RAG is a model adaptation technique that enhances the performance and contextual relevance of responses from LLM Applications, by combining pre-trained language models with external knowledge sources. Retrieval Augmentation uses vector mechanisms and embedding.'**
>
> OWASP

In short, the goal is to provide the model with relevant, current context so it is less reliant on its training data alone, reducing, but not eliminating, the risk of hallucination.

> **'Large pre-trained language models have been shown to store factual knowledge in their parameters, and achieve state-of-the-art results when fine-tuned on downstream NLP tasks. However, their ability to access and precisely manipulate knowledge is still limited.'**
>
> Lewis et al

Why making your LLMs more useful can have the opposite effect

It is worth considering that RAG systems not only rely on the quality of the retrieved data, but also on their ability to validate outputs. That means if the data is poor, so are the outputs. But the real task comes down to locking them down properly.

> **'Vectors and embeddings vulnerabilities present significant security risks in systems utilizing RAG with LLMs. Weaknesses in how vectors and embeddings are generated, stored, or retrieved can be exploited by malicious actions (intentional or unintentional) to inject harmful content, manipulate model outputs, or access sensitive information.'**
>
> OWASP

Training-data poisoning often requires influence over training or fine-tuning data, or over the AI supply chain. RAG poisoning can be lower-friction when the retrieval pipeline ingests attacker-controllable content.

But if we look at RAG poisoning, it is the other way around. In exposed or weakly governed retrieval pipelines, a threat actor may not need to touch the model itself. They may only need to place crafted content somewhere the RAG system ingests, indexes, and later retrieves.

This could be a single public web page or even a wiki page. If added, the malicious communication will be scraped along with everything else. And because LLMs trust what they are given, the model may follow the malicious instruction in its response. In agentic systems with connected tools or permissions, poisoned context can also steer downstream actions.

Would you know how to secure your RAG system from data poisoning?

Because RAG systems retrieve top-ranked content from indexed sources, threat actors may try to make poisoned content appear highly relevant to likely user queries. So, to make sure a poisoned document is among the data collected, threat actors will use techniques such as semantic optimization.

This is when highly relevant semantic keywords are used. Threat actors may use retrieval content crafting: writing or hiding content so it ranks highly for specific queries, including through repeated keywords, semantically related phrases, metadata, or visually hidden instructions that are readable when the system and vectorizers search for them. If the keywords rank as the best match, based on the semantics, it is fed into the LLM's prompt window.

What makes these attacks particularly effective is that traditional firewalls and WAFs are unlikely to catch this reliably. They are built around network traffic, signatures, URLs, and payload patterns, not around whether a legitimate-looking document contains instructions that will later poison a retrieval pipeline. Whereas RAG poisoning is text-based, which means it is written in plain language. The malicious instructions are written within the semantic meaning of the text, not in the code itself.

> **'The content may be targeted such that it would always surface as a search result for a specific user query. The adversary's content may include false or misleading information. It may also include prompt injections with malicious instructions, or false RAG entries.'**
>
> MITRE

AI tools that use RAG are extremely useful to companies for information retrieval and asset creation but can be harmful in the wrong hands.

Business impact when harmful instructions are saved into the AI's memory

Business consequences of successful poisoning can be significant. Data leakage, unauthorized workflow actions, and damage to customer trust are just a handful of ways a business can be impacted.

> **'Poisoned RAG could achieve a 90% attack success rate when injecting five malicious texts for each target question into a knowledge database with millions of texts.'**
>
> Researchers Zou, Geng, Wang, and Jia

While this was a research result under a defined experimental setup, and not a general guarantee that all enterprise RAG systems can be poisoned with five malicious texts, they also evaluated several defenses, all of which were insufficient to defend against poisoned RAG, highlighting the need for new defenses.

Why context persistence matters

A related class of attacks, persistent prompt injection through long-term memory, shows why context persistence matters. In 2024, Embrace The Red demonstrated a ChatGPT macOS proof of concept dubbed 'spAIware', where malicious instructions could be stored in memory and survive across conversations.

> **'Malicious instructions were injected into the long-term memory that survived across chat sessions via memory RAG context.'**
>
> Gulyamov et al
>
> PROMPT INJECTION ATTACKS IN LARGE LANGUAGE MODELS AND AI AGENT SYSTEMS: A COMPREHENSIVE REVIEW OF VULNERABILITIES, ATTACK VECTORS, AND DEFENSE MECHANISMS

When harmful instructions get saved into the AI's memory through a normal-looking conversation, they can impact future chats. This can continue even after you close the session, log out, or switch devices, because the memory is stored on the server.

Memory features are used to make AI feel more personal to you by remembering your preferences and past conversations. But here they were misused.

> **'This attack chain demonstrates the dangers of having long-term memory being automatically added to a system, both from a misinformation/scam point of view, but also regarding continuous communication with attacker-controlled servers. The proof-of-concept demonstrated that this can lead to persistent data exfiltration, and technically also to establish a command-and-control channel to update instructions.'**
>
> Embrace The Red

Securing your pipeline against poisoning

Traditional security measures alone are no longer enough to discover developing vulnerabilities such as RAG poisoning. Legacy tools weren't built to spot a poisoned document sliding unnoticed into a retrieval pipeline. They weren't designed to catch vulnerabilities like this.

> **Securing against RAG poisoning is different for every company; it depends on how your system is built, how retrieved content is handled, and what the system is actually meant to do, since that purpose is what turns an odd output into a finding that matters. There's no quick fix, and because every setup is unique, the gaps that matter are specific to yours. Here, bug bounty can really help, with a diverse pool of researchers continuously probing the system.**
>
> **Lennaert Oudshoorn, Head of Triage, Intigriti**

That's why Intigriti's AI Security and Safety program exists as a scoped, managed program that directs real AI-fluent researchers at your attack surface. These are specialist researchers with experience in retrieval pipelines, embedding manipulation, indirect prompt injection, and agentic abuse cases. They read the academic papers, and they execute the exploits.

Our AI-fluent researchers help surface the gaps that conventional testing often misses.

If you're serious about testing your AI the way an actual threat actor would, not the way a compliance checklist imagines, then it's time to talk.

AUTHOR

Eleanor Barlow

Eleanor Barlow is a London-based Senior Cyber Security Technical Writer at Intigriti, with 9+ years' experience reporting on and writing for the cyber and tech sector. She specializes in data-driven content on cybersecurity and bug bounty intelligence, helping organizations benefit from the latest trends and insights.