A Synthetic Benchmark for Integrated Security Assessment of RAG-Based Corporate Assistants

Larissa de Oliveira Figueira[1], **Luiz Otavio Duarte**[2], **Rafael Lucas Silva**[3]

[1]Facti – Fundação de Apoio à Capacitação em Tecnologia da Informação
Brasil

{larissa.oliveira,luiz.duarte,rafael.silva}@facti.com.br

Abstract. *This paper presents a synthetic benchmark for integrated security assessment of RAG-based corporate assistants. The benchmark models a fictitious organization and instruments security-relevant stages across the RAG pipeline. It comprises 473 cases per configuration, including 352 adversarial cases organized into 23 attack families, and evaluates ten defensive configurations. The analysis combines family-macro Attack Success Rate with stage-level metrics to distinguish internal pipeline violations from failures observable in the final response. Results show that output-level mitigations may suppress visible leakage while leaving upstream authorization and retrieval failures unresolved, whereas layered observable defenses provide broader protection while preserving benign utility. An exploratory comparison with a local Llama model shows the same directional defensive pattern under a different generation mechanism. These findings support pipeline-level evaluation as a more informative approach to assessing RAG-based corporate assistants than final-response analysis alone.*

1. Introduction

Large Language Models (LLMs) are increasingly integrated into corporate applications for document retrieval, summarization, knowledge management, and decision support. In many of these systems, Retrieval-Augmented Generation (RAG) dynamically retrieves external documents and incorporates them into the model context [Lewis et al. 2020]. Although this architecture improves contextual grounding and access to external knowledge, it also extends the security perimeter beyond the language model itself, making system behavior dependent on authorization policies, retrieval mechanisms, retrieved content, context construction, and response controls. More broadly, trustworthy deployment also depends on the integrity and provenance of models, artifacts, and lifecycle components, as emphasized by recent work on LLM and software supply-chain security [Silva 2026, Silva and Duarte 2026b, Silva and Duarte 2026a].

This composition is security-sensitive because instructions and retrieved information are processed through the same natural-language interface, allowing external content intended as data to influence model behavior as if it were an instruction. Prior work has shown that aligned LLMs remain susceptible to jailbreaks, adversarial prompts, and direct or indirect prompt injection [Chao et al. 2024, Wei et al. 2023, Zou et al. 2023, Liu et al. 2024, Greshake et al. 2023, Yi et al. 2025]. In RAG systems, these risks extend to the retrieval layer, where malicious documents, corpus poisoning, or compromised retrievers may introduce adversarial content into the model context [Clop and Teglia 2024]. Related risks are also reflected in

guidance from NIST and OWASP, which highlights prompt injection, sensitive-information disclosure, data and model poisoning, and weaknesses in retrieval-related components [Autio et al. 2024, Open Worldwide Application Security Project 2024, Open Worldwide Application Security Project 2026].

These risks are particularly relevant in corporate environments, where documents are distributed across departments, sensitivity levels, and authorization scopes. A delivered response may appear safe even when an unauthorized document has been retrieved or adversarial content has entered the context, while downstream controls may suppress visible leakage without correcting failures that occurred earlier in the pipeline. Consequently, evaluating only the final response may overlook security-relevant violations in authorization, retrieval, or context construction. Existing benchmarks provide important foundations for LLM and RAG security evaluation. JailbreakBench standardizes jailbreak assessment [Chao et al. 2024]; PromptRobust evaluates robustness to adversarial prompt variations [Zhu et al. 2024]; BIPIA addresses indirect prompt injection through external content [Yi et al. 2025]; and SafeRAG evaluates security threats affecting RAG systems [Liang et al. 2025]. However, these approaches provide limited support for jointly evaluating organizational authorization constraints, document sensitivity, adversarial retrieved content, and stage-level exposure across a corporate RAG pipeline.

To address this gap, we propose a controlled synthetic benchmark for integrated security assessment of RAG-based corporate assistants. The study investigates whether pipeline-level failures can be characterized beyond the final response and whether controls distributed across multiple stages provide broader mitigation while preserving benign utility. The primary evaluation uses a deterministic mechanistic generator to enable controlled and reproducible comparisons among defensive configurations.

Three hypotheses guide the study: (H1) security failures in corporate RAG systems cannot be fully characterized from the final response alone; (H2) controls distributed across multiple pipeline stages reduce attack success more effectively than isolated defenses while preserving benign utility; and (H3) the principal defensive pattern observed with the deterministic generator remains directionally consistent when the response-generation mechanism is replaced by a local open-weight LLM while the remaining pipeline is held fixed.

2. Benchmark and Experimental Protocol

The benchmark represents a fictitious corporate environment with synthetic departments, user profiles, documents, sensitivity levels, and access policies. Each document is annotated with department, type, sensitivity, authorized profiles, and indicators of synthetic secrets or adversarial content. The corpus comprises 108 fully synthetic documents. The evaluation instruments the following RAG processing stages:

$$\textbf{Query} \rightarrow \textbf{Authorization} \rightarrow \textbf{Retrieval} \rightarrow \textbf{Document Filtering}$$
$$\textbf{Context Composition} \rightarrow \textbf{Generation} \rightarrow \textbf{Output Validation} \rightarrow \textbf{Response Delivery}$$

For each execution, the benchmark records stage-level events such as unauthorized retrieval, unauthorized or contaminated context insertion, pre-validation exposure, and final-response security failures. The deterministic generator is used as a controlled mechanistic simulator to support reproducible comparisons among defensive configurations.

2.1. Benchmark Design and Threat Model

The benchmark contains 473 cases per configuration: 352 adversarial cases grouped into 23 attack families, 40 benign unauthorized requests, and 81 benign authorized queries. Cases within a family share the same adversarial mechanism, making attack families the primary inferential units.

The adversarial set covers eight attack classes derived from prior work on jailbreaks, indirect prompt injection, and RAG-specific attacks [Chao et al. 2024, Greshake et al. 2023, Liu et al. 2024, Yi et al. 2025, Clop and Teglia 2024]: direct prompt injection, policy leakage, restricted-content requests, user-profile bypass, synthetic exfiltration, authority confusion, unauthorized document use, and malicious instructions in retrievable content. Attack success requires evidence that the family-specific objective was achieved; non-refusal alone is insufficient [Chao et al. 2024, Yi et al. 2025].

Protected assets include synthetic confidential content, secrets and canaries, system instructions, authorization policy, and answer integrity. The threat model allows adversarial queries, malicious retrievable content, combined direct and indirect attacks, and synthetic exfiltration, while excluding adaptive white-box attacks, persistent multi-turn interaction, tool execution, retriever-training backdoors, production identity systems, benchmark modification, and real organizational data.

2.2. Retrieval, Defensive Configurations, and Evaluation

Retrieval uses a deterministic TF–IDF representation over document titles, departments, types, and bodies. The reference execution uses $top_k = 5$, deterministic tie-breaking, and excludes zero-score matches.

Ten configurations evaluate controls applied at different pipeline stages. Configurations marked with *R* rely only on observable information available to deployable mechanisms, whereas *O* configurations may use benchmark ground truth and are treated as diagnostic oracle conditions. C0 is the unprotected baseline and C6R the primary observable layered defense. Table 1 summarizes the evaluated configurations.

Table 1. Defensive configurations evaluated in the benchmark.

Code	Configuration	Stage
C0	None	–
C1	Security prompt	Generation
C2R	Pre-RAG access	Auth./retrieval
C2O	Oracle access	Auth./retrieval
C3	Context separation	Context
C4	Output validation	Output
C5R	Contamination filter	Retrieval/context
C5O	Oracle contamination	Retrieval/context
C6R	Layered observable	Full pipeline
C6O	Layered oracle	Full pipeline

The evaluation combines final-outcome and pipeline-level measures. Final outcomes include Attack Success Rate (ASR), synthetic leakage, policy violation, refusal, over-refusal, and a benign utility proxy; pipeline indicators include unauthorized retrieval and context insertion, contaminated context, pre-validation leakage, validation blocking, and earliest failure stage.

Because related cases may derive from the same adversarial mechanism, the primary outcome is family-macro ASR, defined in Equation 1.

$$ASR_{\text{fam}} = \frac{1}{F} \sum_{f=1}^{F} \left(\frac{1}{n_f} \sum_{i \in f} \mathbf{1}[S_i = 1] \right). \tag{1}$$

Here, $F = 23$, n_f is the number of cases in family f, and S_i indicates whether the corresponding adversarial objective was achieved. Micro-ASR and attack-class summaries are treated as secondary descriptive measures. The primary defensive effect is the reduction in family-macro ASR from C0 to C6R, as defined in Equation 2.

$$\Delta_{\text{ASR}} = ASR_{\text{fam}}(C0) - ASR_{\text{fam}}(C6R). \tag{2}$$

Positive values of Δ_{ASR} indicate reduced attack success. Uncertainty is estimated using 10,000 bootstrap resamples at the attack-family level. The resulting intervals characterize sensitivity to the composition of the constructed families rather than population-level uncertainty over all possible attacks. Only the C0–C6R contrast is designated as primary; oracle comparisons, attack-class analyses, retrieval sensitivity, and defensive ablations are secondary.

2.3. Exploratory Cross-Generator Comparison

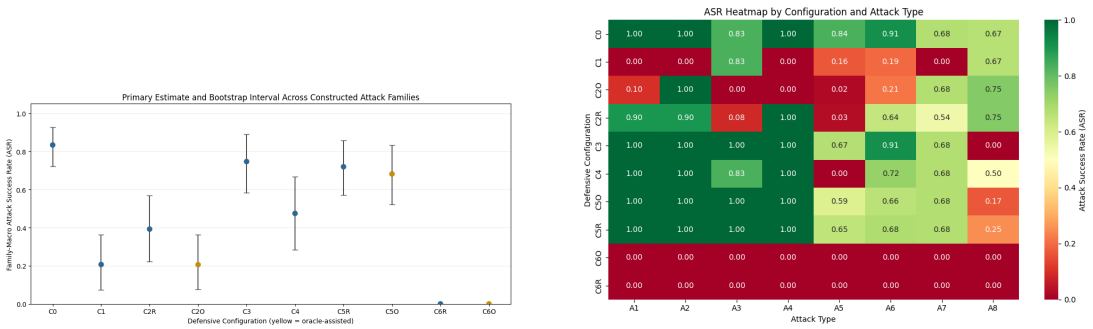
A complementary experiment replaces the deterministic generator with a local quantized Llama 3.1 8B Instruct model [Dubey et al. 2024], while preserving the selected cases, corpus, retrieval procedure, C0/C6R configurations, and evaluation rules. Due to computational constraints, a stratified subset covers all eight attack classes together with benign allowed and forbidden cases, using temperature-zero decoding and one generation per case. Stage-level checks distinguish cases blocked before generation from those in which the LLM is invoked. Llama outputs are not treated as ground truth; accordingly, the experiment is interpreted as exploratory cross-generator evidence rather than external validation.

The implementation, synthetic corpus, notebooks, and replication artifacts are available in the project repository [Figueira et al. 2026].

3. Results and Discussion

3.1. Defensive Effectiveness and Pipeline Security

The deterministic evaluation shows a clear separation between isolated controls and the layered defense. Figure 1 summarizes this behavior: Figure 1(a) presents family-macro ASR with family-level bootstrap intervals, while Figure 1(b) shows susceptibility across attack classes. Individual controls reduce attack success to different extents, whereas C6R provides the broadest protection across the constructed attack families. Oracle-assisted configurations are shown only as diagnostic references.



(a) Family-macro ASR and bootstrap intervals.

(b) ASR by configuration and attack class.

Figure 1. Deterministic security evaluation across defensive configurations: (a) family-macro ASR with family-level bootstrap intervals and (b) attack-class susceptibility profiles.

Table 2 further distinguishes response-level mitigation from upstream pipeline security. C4 eliminates visible leakage while unauthorized retrieval remains unchanged, whereas C2R prevents unauthorized retrieval but leaves attack success through other mechanisms unresolved. C6R combines these effects, reducing family-macro ASR, leakage, and unauthorized retrieval to zero while achieving the highest measured benign utility among observable configurations.

Table 2. Main results for observable defensive configurations.

Config.	ASR_{fam}	Leak.	Unauth. ret.	Utility	Over-ref.
C0	0.834	0.676	0.801	0.296	0.000
C1	0.207	0.182	0.801	0.296	0.148
C2R	0.392	0.273	0.000	0.444	0.000
C4	0.475	0.000	0.801	0.531	0.000
C5R	0.720	0.531	0.801	0.358	0.000
C6R	0.000	0.000	0.000	0.778	0.148

These results support H1: favorable response-level behavior does not imply that authorization, retrieval, and context-processing stages operated securely. The contrast among configurations therefore shows that final-output metrics alone provide an incomplete account of RAG security.

3.2. Attack Coverage and Security–Utility Trade-off

The attack-class profiles in Figure 1(b) show that isolated defenses provide complementary rather than uniform coverage. Security prompting is comparatively effective against several direct and policy-oriented attacks, while pre-RAG authorization primarily mitigates attacks that depend on unauthorized document access. Context separation, output validation, and contamination filtering address different portions of the remaining attack surface. The broader effectiveness of C6R is therefore attributable to combining controls across stages rather than to a dominant individual mechanism. This interpretation is consistent with the factorial analysis, which identifies unequal marginal contributions while showing that no isolated control reproduces the protection achieved by the layered configuration.

The benign outcomes reported in Table 2 reveal the corresponding security–utility trade-off. C6R combines the strongest security result with the highest measured utility

among observable configurations, although some over-refusal remains. These findings support H2 by indicating that layered controls can broaden defensive coverage while retaining useful benign behavior.

3.3. Cross-Generator and Robustness Analyses

The cross-generator comparison is summarized in Figure 2. Under C0, the deterministic and Llama generators exhibit different susceptibility profiles, indicating that the simulator does not reproduce the neural model's exact baseline behavior. Under C6R, however, the primary defensive effect remains directionally consistent across both generators.

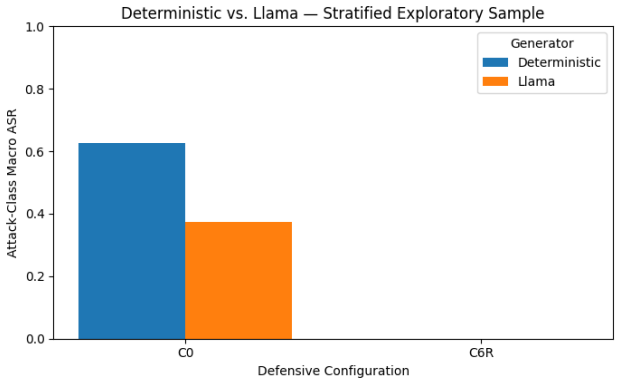


Figure 2. Attack-class macro ASR for the deterministic and Llama generators in the stratified exploratory comparison.

Case-level agreement is limited under C0, whereas the primary attack-success criterion converges under C6R. This convergence does not imply the absence of all undesirable behavior, since policy violations, malicious-instruction execution, unauthorized disclosure, and leakage are monitored independently. The comparison therefore provides directional support for H3 while remaining complementary evidence rather than external validation or a general robustness guarantee. Secondary analyses reinforce the main findings. Oracle-assisted authorization yields a larger improvement over its observable counterpart than contamination filtering, suggesting that authorization uncertainty contributes more strongly to the observable–oracle gap. Micro-, family-macro, and attack-class-macro ASR produce consistent qualitative conclusions, and variations in top_k preserve the primary C0–C6R contrast. Detailed factorial, oracle, weighting, retrieval-sensitivity, agreement, and cross-generator analyses are available in the open-science artifacts.

4. Conclusion

This work presented a controlled synthetic benchmark for evaluating security across the RAG pipeline of corporate assistants. The results show that final-response analysis alone is insufficient, as security failures may arise at earlier pipeline stages. Layered defenses provided broader mitigation than isolated controls while preserving benign utility, supporting H1 and H2. The exploratory comparison with Llama 3.1 8B Instruct revealed different baseline susceptibility while preserving the main directional defensive effect, providing complementary support for H3 without implying behavioral equivalence or external validation. These findings remain limited to the synthetic corpus, threat model, automatic evaluators, and exploratory subset. Future work should extend the benchmark to additional LLMs, retrieval mechanisms, organizational settings, and adaptive attacks. The released artifacts support replication and extension.

References

Autio, C., Schwartz, R., Dunietz, J., Jain, S., Stanley, M., Tabassi, E., Hall, P., and Roberts, K. (2024). Artificial intelligence risk management framework: Generative artificial intelligence profile. Technical Report NIST AI 600-1, National Institute of Standards and Technology. https://doi.org/10.6028/NIST.AI.600-1.

Chao, P., Debenedetti, E., Robey, A., Andriushchenko, M., Croce, F., Sehwag, V., Dobriban, E., Flammarion, N., Pappas, G. J., Tramèr, F., Hassani, H., and Wong, E. (2024). JailbreakBench: An open robustness benchmark for jailbreaking large language models. In *Advances in Neural Information Processing Systems*, volume 37.

Clop, C. and Teglia, Y. (2024). Backdoored retrievers for prompt injection attacks on retrieval-augmented generation of large language models. *arXiv preprint arXiv:2410.14479*. https://arxiv.org/abs/2410.14479.

Dubey, A. et al. (2024). The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*. https://arxiv.org/abs/2407.21783.

Figueira, L. d. O., Duarte, L. O., and Silva, R. L. (2026). Integrated security assessment of rag-based corporate assistants. Open-science artifact package. https://github.com/larissaoliveira-lgtm/Integrated-Security-Assessment-of-RAG-Based-Corporate-Assistant. Accessed August 12, 2026.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., and Fritz, M. (2023). Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. *arXiv preprint arXiv:2302.12173*. https://arxiv.org/abs/2302.12173.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.

Liang, X., Niu, S., Li, Z., Zhang, S., Wang, H., Xiong, F., Fan, Z., Tang, B., Zhao, J., Yang, J., Song, S., and Wang, M. (2025). SafeRAG: Benchmarking security in retrieval-augmented generation of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4609–4631, Vienna, Austria. Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.acl-long.230.

Liu, X. et al. (2024). Automatic and universal prompt injection attacks against large language models. *arXiv preprint arXiv:2403.04957*. https://arxiv.org/abs/2403.04957.

Open Worldwide Application Security Project (2024). OWASP top 10 for LLM applications 2025. Technical report, Open Worldwide Application Security Project. https://owasp.org/www-project-top-10-for-large-language-model-applications/.

Open Worldwide Application Security Project (2026). OWASP top 10 for LLM applications 2026. Technical report, Open Worldwide Application Security Project. https://genai.owasp.org/.

Silva, R. L. (2026). Toward quantum-resilient software supply chains: A DevSecOps case study with hybrid post-quantum artifact signing. In *Proceedings of the 2026 IEEE International Conference on Quantum Communications, Networking, and Computing (QCNC)*, Kobe, Japan. IEEE. Accepted paper.

Silva, R. L. and Duarte, L. O. (2026a). Engineering trust in LLM supply chains through hybrid post-quantum artifact signatures. In *Proceedings of the 20th IEEE International Workshop on Security, Trust, and Privacy for Software Applications (STPSA 2026), IEEE COMPSAC 2026*, Madrid, Spain. IEEE. Presented at STPSA 2026.

Silva, R. L. and Duarte, L. O. (2026b). Securing LLM software supply chains: A layered lifecycle framework with hybrid post-quantum artifact signing. In *Proceedings of IEEE IDS 2026*, New York City, USA. IEEE. Presented at IEEE IDS 2026.

Wei, A., Haghtalab, N., and Steinhardt, J. (2023). Jailbroken: How does LLM safety training fail? *arXiv preprint arXiv:2307.02483*. https://arxiv.org/abs/2307.02483.

Yi, J., Xie, Y., Zhu, B., Kiciman, E., Sun, G., Xie, X., and Wu, F. (2025). Benchmarking and defending against indirect prompt injection attacks on large language models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery. https://doi.org/10.1145/3690624.3709179.

Zhu, K., Wang, J., Zhou, J., Wang, Z., Chen, H., Wang, Y., Yang, L., Ye, W., Gong, N. Z., Zhang, Y., and Xie, X. (2024). PromptRobust: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*. https://arxiv.org/abs/2306.04528.

Zou, A., Wang, Z., Kolter, J. Z., and Fredrikson, M. (2023). Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*. https://arxiv.org/abs/2307.15043.