Securing Retrieval-Augmented Generation (RAG) with Trusted Execution Environments: A Comparative Security Evaluation

Problem Statement and Goals

Problem Statement

- Retrieval-Augmented Generation (RAG) systems are increasingly used over private and sensitive document collections.
- Despite their benefits, RAG pipelines introduce new privacy and security risks, including membership inference, data poisoning, prompt injection, and query pattern inference attacks.
- Trusted Execution Environments (TEEs) are proposed as a defense, but their real-world effectiveness and limitations in RAG systems remain insufficiently evaluated.

Goals

- Systematically evaluate security and privacy risks across representative RAG settings.
- Analyze the effectiveness of TEE isolation and lightweight software defenses against realistic attacks.
- Use a comparative evaluation framework to identify practical defense combinations that balance security and performance.

Approach

- Developing a comparative evaluation framework across representative RAG settings.
- Modeling realistic privacy attacks, including membership inference, data poisoning, prompt injection, and query pattern inference.
- Implementing a TEE-enabled RAG pipeline to enforce isolation of security-critical components.
- Evaluating individual and combined defenses under identical attack scenarios.
- Measuring security and utility to analyze trade-offs.

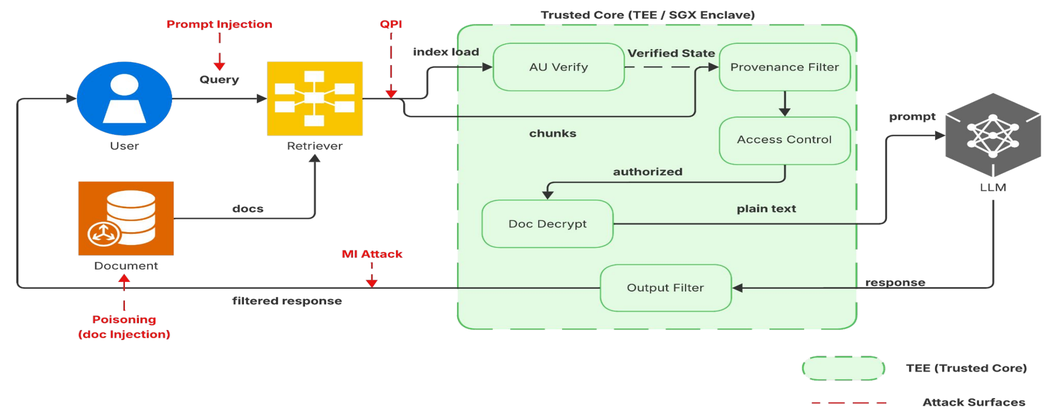

Fig 1. TEE-Enabled RAG Pipeline with Trusted Boundary

Results

Results and Insights

- Baseline RAG is vulnerable: Poisoning ASR 43.5%, Prompt Injection PII leakage 4%, MI AUC 0.983.
- TEE bundle eliminates poisoning (43.5%→0%), injection leakage (4%→0%), and reduces MI to near-random (AUC 0.983→0.052), with no answer quality degradation (ΔEM ≈ 0pp).
- Defense effectiveness is attack-specific: Provenance and AU dominate for poisoning; ACL is critical for MI (AUC 0.983→0.052); Output Filter and ACL block injection PII leakage; QPI remains unmitigated.
- Limitations: Prompt injection PII leakage (4%) is low due to Llama 3.1 70B's safety alignment; weaker models may show higher rates. QPI remains unmitigated.

Table 1. Individual Defense Effectiveness

Attack \ Defense	Provenance Filter	Attested Updates	Doc Encryption	Output Filter	Access Control
Data Poisoning (ASR)	43.5% → 0% (Strong)	43.5% → 0% (Strong)	43.5% → 43.5% (Limited)	43.5% → 41.5% (Moderate)	43.5% → 43.5% (Limited)
Injection (ASR)	4% → 4% (Limited)	4% → 4% (Limited)	4% → 4% (Limited)	4% → 0% (Strong)	4% → 0% (Strong)
MI (AUC)	0.983 → 0.983 (Limited)	0.983 → 0.983 (Limited)	0.983 → 0.983 (Limited)	0.983 → 0.983 (Limited)	0.983 → 0.052 (Strong)
QPI (Accuracy)	54.9% → 54.9% (Limited)	54.9% → 54.9% (Limited)	54.9% → 54.9% (Limited)	54.9% → 54.9% (Limited)	54.9% → 54.9% (Limited)

Table 2. Combined Defense Result

Metric	Baseline	Bundle*	ΔEM
Poisoning	43.5%	0%	≈0pp
Injection	4%	0%	≈0pp
MI	0.983	0.052	≈0pp
QPI	54.9%	54.9%	≈0pp

Bundle*: Provenance Filter + AU + Doc enc+ Output Filter + ACL

Jihu Hwang
Carnegie Mellon University

Carnegie Mellon University

Securing Retrieval-Augmented Generation (RAG) with Trusted Execution Environments: A Comparative Security Evaluation

Abstract

Retrieval-Augmented Generation (RAG) systems are increasingly deployed over private and sensitive document collections to improve factual accuracy and reduce hallucination. [1][2][3] However, the retrieval pipeline introduces new security and privacy risks, including membership inference [4], data poisoning [5], and prompt injection attacks [3] that can trigger unintended disclosure or manipulation of retrieved content and generated outputs. [2][3] Trusted Execution Environments (TEEs), such as Intel SGX, are often proposed as a practical defense by isolating security-critical computations from an untrusted host [6][7],yet the real-world effectiveness and limitations of TEE-enabled RAG deployments remain insufficiently characterized.

This project develops a comparative security evaluation framework for TEE-enabled RAG systems. I study three baseline configurations: a large language model (LLM) only system without retrieval, a BM25-based RAG pipeline, and a dense retriever-based RAG pipeline. Using consistent attack scenarios and metrics across baselines, I evaluate how TEE isolation interacts with practical software controls, including attested updates, access control, document-level encryption, and output filtering. The evaluation focuses on security impact, utility degradation, and performance overhead to capture practical trade-offs.

The expected outcome is a comprehensive attack defense evaluation matrix showing how five defense mechanisms are expected to mitigate four threats across representative RAG settings. By grounding recommendations in measured threat-mitigation effectiveness and security-utility-performance trade-offs, this work aims to provide evidence-based guidance for securing RAG systems under realistic deployment constraints.

Reference:
[1] https://arxiv.org/abs/2005.11401
[2] https://arxiv.org/abs/2402.16893
[3] https://arxiv.org/abs/2509.20324
[4] https://arxiv.org/abs/2405.20446
[5] https://arxiv.org/abs/2402.07867
[6] https://arxiv.org/abs/2004.05703
[7] https://arxiv.org/abs/1806.03287

Jihu Hwang
Carnegie Mellon University

Carnegie Mellon University