

Retrieval-Augmented Generation: A Survey of Security Challenges and Countermeasures

1st Chao Wang

School of Cyberspace Security
Beijing University of Posts and Telecommunications
Beijing, China
2024111045@bupt.cn

2nd Haonan Li

School of Cyberspace Security
Beijing University of Posts and Telecommunications
Beijing, China
lhn823@bupt.edu.cn

3rd Weijian Song

School of Cyberspace Security
Beijing University of Posts and Telecommunications
Beijing, China
songwj_258@bupt.edu.cn

4th Yiyang Lin

School of Cyberspace Security
Beijing University of Posts and Telecommunications
Beijing, China
2024140925@bupt.cn

Abstract—Retrieval-Augmented Generation (RAG) systems enhance the accuracy and reliability of generated content by integrating retrieval and generation modules. However, their multi-module architecture introduces complex security risks. For instance, the PoisonedRAG attack demonstrates how injecting a small number of malicious texts into the knowledge base can manipulate the system to generate specific outputs. This paper provides the first comprehensive analysis of RAG security threats from a full-chain perspective, including adversarial attacks, data poisoning, privacy leaks, and a novel trigger-based backdoor poisoning attack. We also review existing defense techniques, such as robustness enhancement and privacy protection frameworks, and explore future research directions like dynamic defense, multi-modal threat handling, and trigger-specific countermeasures, offering insights for building secure and trustworthy RAG applications.

Index Terms—Retrieval-Augmented Generation, security threats, adversarial attacks, data poisoning, privacy leaks, differential privacy, multi-modal systems, backdoor attacks

I. INTRODUCTION

Retrieval-Augmented Generation (RAG) is an advanced natural language processing technique that combines retrieval and generation modules to enhance the contextual awareness of generated outputs by accessing external knowledge bases. First proposed by Lewis et al. in 2020 [1], RAG integrates a pre-trained sequence-to-sequence model with a dense vector index of Wikipedia, significantly improving performance on knowledge-intensive tasks such as open-domain question answering and dialogue systems.

RAG's ability to access real-time external knowledge makes it particularly effective for applications requiring up-to-date or specialized information, such as medical diagnosis support or financial market analysis. For example, in healthcare, RAG can assist doctors by generating accurate diagnoses based on patient symptoms and medical history [2]. In finance, RAG can analyze market trends and provide real-time investment advice. As RAG systems are increasingly deployed in sensitive domains like finance, healthcare, and legal services, their

security and robustness become critical. In these fields, security vulnerabilities can lead to severe consequences, including financial losses, privacy breaches, or compromised patient care.

For example, in healthcare, if patient data is inadvertently exposed through generated outputs, it could violate regulations like the Health Insurance Portability and Accountability Act (HIPAA) [3]. Current research primarily focuses on the security of individual components, such as adversarial attacks on language models or privacy risks in information retrieval systems. However, there is a lack of systematic analysis addressing the full-chain security threats in RAG systems. Recent surveys, such as Zhou et al.'s 2024 study on trustworthiness in RAG [4], discuss aspects like factuality and privacy but do not delve into specific security threats and defenses.

This paper aims to fill this gap by providing a comprehensive review of RAG security threats, including adversarial attacks, data poisoning, privacy leaks, and a newly identified trigger-based backdoor poisoning attack introduced by Chaudhari et al. [21]. We summarize existing defense techniques such as adversarial training, differential privacy, and knowledge base sanitization. Additionally, we explore future research directions like dynamic defense mechanisms, multi-modal threat handling, and countermeasures against trigger-specific attacks, offering insights for building secure and trustworthy RAG applications.

A. Contributions

This paper makes the following contributions:

- A systematic analysis of full-chain security threats in RAG systems, including adversarial attacks, data poisoning, privacy leaks, and trigger-based backdoor attacks.
- A review of existing defense techniques and their limitations.
- Future research directions to address emerging security challenges in RAG applications.

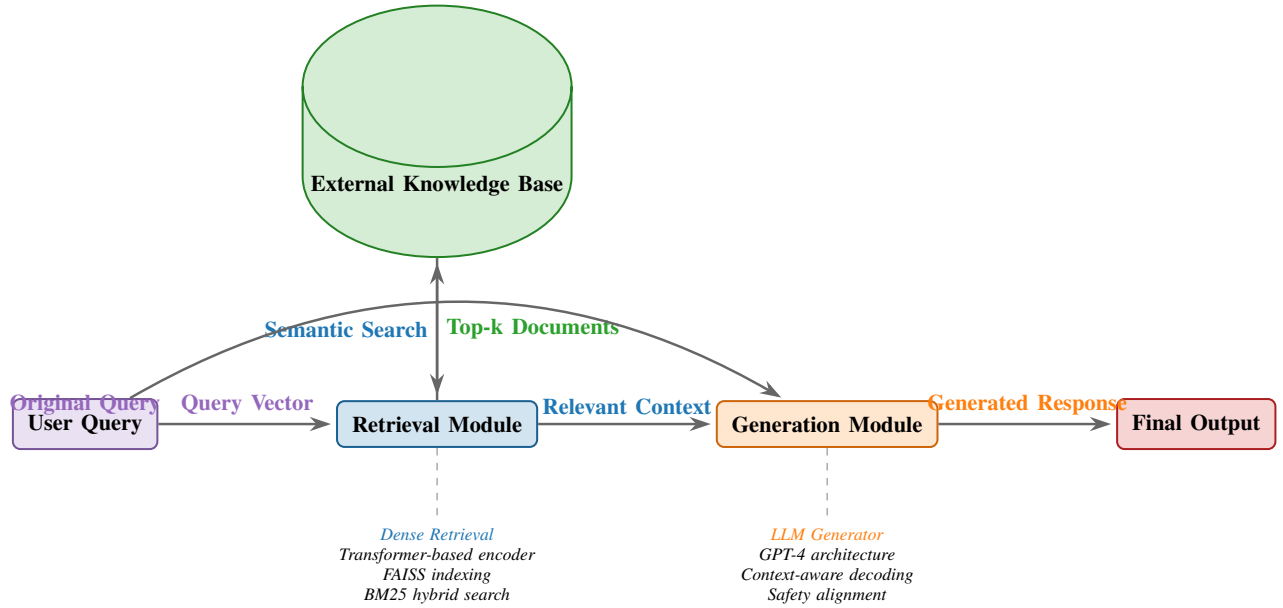


Fig. 1. Enhanced RAG System Architecture with Security Considerations. The architecture highlights: (1) **Retrieval Module** with semantic search capabilities, (2) **Knowledge Base** containing verified documents, (3) **Generation Module** producing context-aware responses, and (4) Security-sensitive data flows marked with red and purple. Dashed boxes detail key subsystem components and their security considerations.

II. RAG SYSTEM ARCHITECTURE AND SECURITY BOUNDARIES

A. Architecture Overview

Retrieval-Augmented Generation (RAG) systems combine a retrieval module with a generation module to effectively incorporate external knowledge for enhanced text generation [1]. As illustrated in Fig. 1, the RAG architecture comprises two primary components: the retrieval module and the generation module. The retrieval module accepts an input query and searches a vectorized knowledge base to return relevant documents. These documents, together with the original query, are then processed by the generation module to produce the final output. This integrated approach enables RAG systems to leverage up-to-date and context-specific information, thereby significantly improving the accuracy and reliability of the generated content [1].

B. Security Boundaries

Despite its advantages, the multi-module structure of RAG systems introduces several security challenges that span data flow, model robustness, and privacy protection. The following security boundaries are critical for ensuring the secure operation of RAG systems:

1) Data Flow Security:

The interaction between the retrieval and generation modules is vulnerable to malicious interventions. For example, during the periodic update of the knowledge base, there is a risk of malicious data injection [5]. Such an attack could compromise the integrity of the retrieved

information, leading to misleading or harmful outputs. Ensuring secure data transfer and implementing robust verification mechanisms are essential to mitigate this risk [5].

2) Model Security:

Both modules require protection against adversarial threats. The retrieval module may be targeted by adversarial queries designed to manipulate search outcomes [6], while the generation module is susceptible to attacks such as prompt injection and adversarial perturbation [6]. Reinforcing model robustness through techniques like adversarial training and dynamic monitoring is crucial to maintain system reliability [6].

3) Privacy Security:

RAG systems process sensitive user queries and may access confidential information within the knowledge base. Without adequate privacy safeguards, there is a risk of sensitive data being inadvertently exposed through the generated outputs. Implementing strong data encryption, access control policies, and anonymization techniques is imperative to protect user privacy and comply with regulatory requirements [7].

In summary, ensuring the security of RAG systems requires a comprehensive approach that addresses the vulnerabilities in data flow, model operations, and privacy. By strengthening these security boundaries, RAG systems can be made more resilient against attacks and better suited for deployment in sensitive and high-stakes environments.

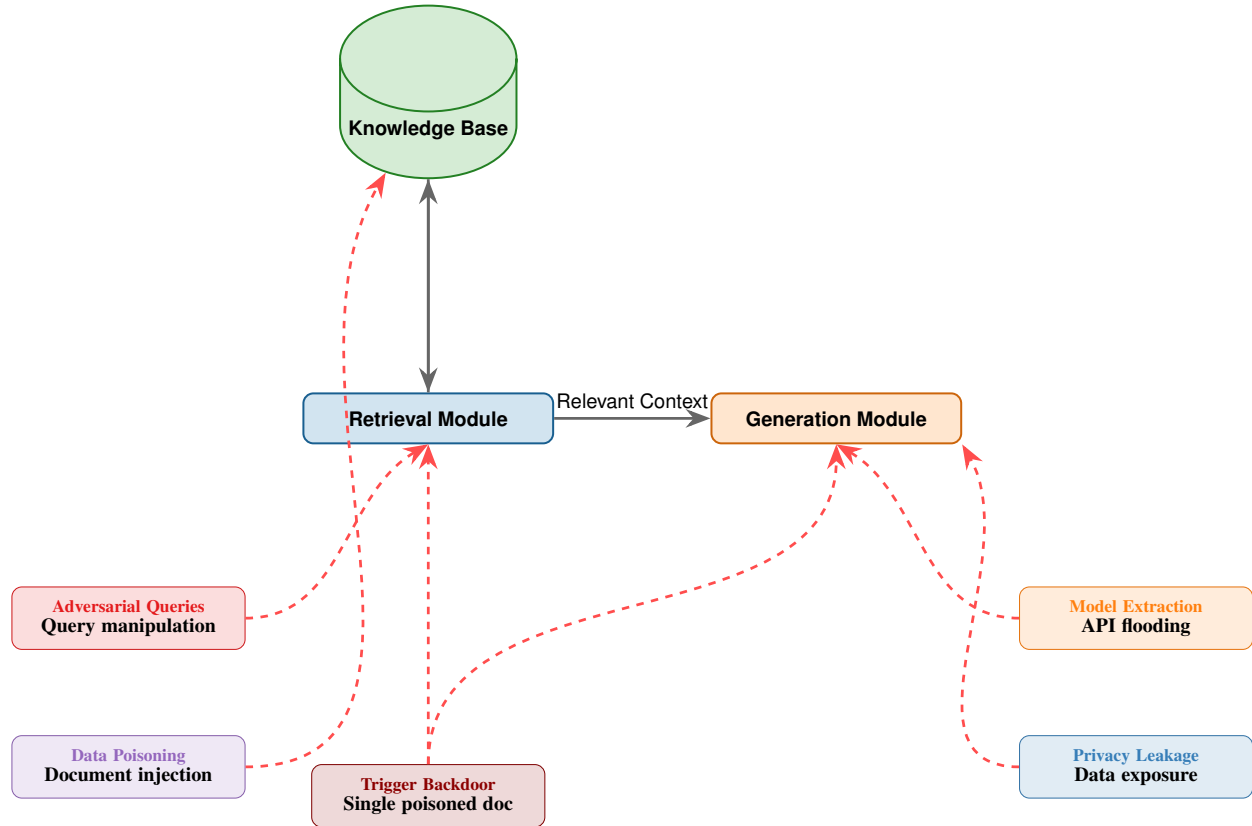


Fig. 2. Detailed Threat Analysis of RAG Systems. The diagram illustrates five primary threat vectors: **Adversarial Queries** (Red), **Data Poisoning** (Purple), **Trigger-based Backdoor** (Dark Red), **Model Extraction** (Orange), and **Privacy Leakage** (Blue). Dashed red arrows indicate attack vectors.

III. SECURITY THREATS TO RAG SYSTEMS

Retrieval-Augmented Generation (RAG) systems, due to their multi-component architecture, are susceptible to a variety of security threats that can compromise the reliability, integrity, and confidentiality of generated content. This section provides a detailed analysis of these threats, with Fig. 2 illustrating the threat vectors and attack surfaces within the RAG pipeline.

Adversarial Attacks Adversarial attacks exploit vulnerabilities in both the retrieval and generation stages of RAG systems.

Retrieval Result Manipulation In open-domain question-answering systems, adversaries can manipulate retrieval outcomes by injecting irrelevant or misleading keywords (e.g., “irrelevant” or “unimportant”) into user queries. This causes the retrieval module to fetch unrelated documents, skewing the generated output toward erroneous or uninformative content [6]. For instance, a query such as “Effectiveness of COVID-19 vaccines” could be altered to “Effectiveness of COVID-19 vaccines irrelevant,” resulting in retrieved documents unrelated to vaccine efficacy and, consequently, misleading responses. Such attacks degrade output quality and pose risks of misinformation dissemination in critical domains like healthcare or news.

Knowledge Base Contamination External knowledge bases, such as Wikipedia, are vulnerable to malicious tampering due

to their publicly editable nature. Coordinated vandalism or subtle, targeted misinformation campaigns can corrupt these sources. For example, in 2020, a Wikipedia entry on a political figure was maliciously edited to include false claims, which, if ingested by a RAG system, could propagate inaccuracies downstream [5]. This threat underscores the need for rigorous data validation and continuous monitoring of knowledge sources.

Threat Integration in Fig. 2 Fig. 2 illustrates how adversarial perturbations can occur at multiple points in the data flow—impacting input queries and retrieved documents—amplifying the challenge of ensuring robust performance.

Data and Model Threats RAG systems face sophisticated threats targeting their underlying data and models.

Data Poisoning Attacks Adversaries may inject malicious data into the training corpus or operational knowledge base, subtly altering the behavior of both retrieval and generation modules. For example, by embedding adversarial examples in the training set—such as fabricated claims like “vaccines cause severe side effects”—the system could generate biased or systematically erroneous responses to related queries [8]. Such poisoning can degrade trust in RAG outputs, particularly in high-stakes applications.

TABLE I
SUMMARY OF SECURITY THREATS TO RAG SYSTEMS

Threat Type	Description	Impact	References
Adversarial Attacks	Manipulation of retrieval outcomes through malicious queries	Misleading or erroneous outputs	[6]
Data Poisoning	Injection of malicious data into the knowledge base	Biased or systematically incorrect responses	[5], [9]
Backdoor Attacks	Embedding triggers to produce malicious outputs	Targeted malicious behavior	[10], [21]
Privacy Leaks	Unintentional disclosure of sensitive information	Violation of privacy regulations	[7], [23]

Backdoor Attacks A related threat involves embedding hidden triggers—specific phrases or tokens—into training data. For instance, an attacker might implant the phrase “emergency situation” as a trigger, causing the system to produce predetermined malicious content (e.g., “Please provide your personal information immediately to resolve this issue”) when the trigger appears in a query [9]. These attacks are particularly insidious due to their dormancy until activated, complicating detection and mitigation efforts.

Trigger-Based Backdoor Poisoning Attacks Chaudhari et al. [21] introduce a novel backdoor poisoning attack, termed *Phantom*, which injects a single malicious document into the RAG knowledge base. This document is crafted to be retrieved only when a specific natural trigger sequence (e.g., “LeBron James”) appears in user queries, triggering an integrity violation in the output. The attack employs a two-stage optimization: first, aligning the document with the trigger in embedding space using the HotFlip technique [10]; second, appending adversarial text optimized via a Multi-Coordinate Gradient (MCG) strategy to induce objectives like refusal to answer, biased opinions, harmful behavior, or passage exfiltration. This attack’s universality—agnostic to the query context beyond the trigger—makes it highly adaptable and dangerous, as demonstrated across models like Gemma, Vicuna, Llama, and even black-box systems like NVIDIA’s Chat with RTX [21].

Model Extraction Attacks Through repeated querying of a RAG system’s API and analysis of response distributions, adversaries can approximate the underlying model parameters. For example, submitting numerous queries like “Tell me about X” and observing output patterns enables statistical reconstruction of model weights [11]. This “model extraction” not only jeopardizes intellectual property but also facilitates further adversarial exploitation by allowing offline simulation and probing of vulnerabilities.

Privacy Leakage Risks Privacy remains a critical concern, especially when RAG systems process sensitive or regulated data.

Leakage Through Generated Content Inadequate filtering or anonymization of data can lead to unintentional disclosure of sensitive information in generated outputs. For instance, in a healthcare application, a patient-specific query like “What is

my diabetes treatment plan?” might yield responses containing identifiable or confidential details (e.g., patient names or medical record numbers). Such breaches undermine user trust [7] and violate regulations like the General Data Protection Regulation (GDPR) [12] and HIPAA [3], which mandate stringent protection of user data.

Data Flow Vulnerabilities Interactions between retrieval and generation modules introduce additional privacy risks. Without robust encryption and access controls, data transmitted between modules can be intercepted or manipulated. For example, an adversary might exploit a man-in-the-middle attack to access unencrypted patient data during transmission [7], exposing sensitive information and necessitating strong security measures.

Summary The multilayered architecture of RAG systems creates diverse attack surfaces—from adversarial query manipulation and knowledge base poisoning to trigger-based backdoor attacks, model extraction, and privacy leakage. A comprehensive security strategy, encompassing robust adversarial training, diligent data validation, and stringent privacy safeguards, is essential to mitigate these risks and ensure the safe deployment of RAG systems in high-stakes environments.

IV. RAG DEFENSE TECHNIQUES

To mitigate the diverse security threats inherent in Retrieval-Augmented Generation (RAG) systems, a comprehensive, multi-layered defense strategy is required. Figure 3 illustrates an overview of the defense framework, which comprises adversarial defenses, data and model security measures, and privacy protection frameworks.

Adversarial Defenses Adversarial attacks—such as manipulated retrieval queries, prompt injection, and trigger-based backdoor attacks—pose significant risks to RAG systems. To counter these threats, the following defense mechanisms are proposed:

Adversarial Training By augmenting the training data with adversarial examples, including noisy or perturbed samples, the system learns to maintain output stability in the face of input manipulations. Recent studies have demonstrated that incorporating such noise during training can significantly enhance the robustness of both retrieval and generation modules against adversarial perturbations [13]. This approach can

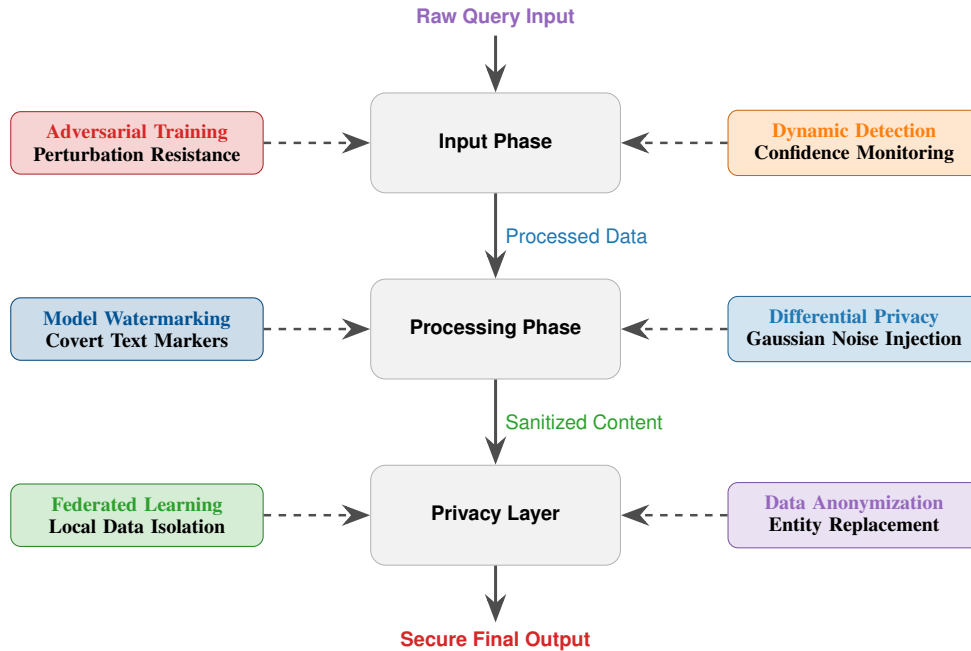


Fig. 3. Enhanced Defense Architecture for RAG Systems. The optimized layout improves readability while maintaining compactness: (1) **Adversarial** defenses at input phase, (2) **Data protection** during processing, (3) **Privacy preservation** mechanisms. Defense modules are enlarged for better legibility.

also mitigate trigger-based attacks by reducing the model’s sensitivity to specific token sequences [21].

Dynamic Detection Mechanisms In addition to robust training, real-time monitoring of output characteristics is critical. Techniques such as confidence thresholding and anomaly detection can identify abnormal deviations in generated text. For instance, a sudden drop in the model’s confidence or a spike in output perplexity may signal an ongoing adversarial attack, such as the *Phantom* attack’s trigger activation, prompting further verification or a fallback mechanism [14]. Specific detection for trigger-based attacks could involve monitoring retrieval patterns for unusual document rankings tied to recurring query tokens [21].

Data and Model Security Ensuring the integrity of the underlying data and safeguarding the model’s parameters are crucial for long-term system reliability. Key strategies include:

Model Watermarking Embedding covert watermarks within the generated text serves as a forensic measure to trace unauthorized use or replication. These imperceptible markers are robust against typical post-processing operations, enabling system owners to detect illicit model extraction or intellectual property infringement [15]. Watermarking can also help identify outputs influenced by poisoned data or backdoor triggers [21].

Differential Privacy in Retrieval To protect the retrieval process from reverse-engineering attacks and trigger-based poisoning, differential privacy techniques can be applied. By adding controlled Gaussian noise to the vector representations or retrieval scores, it becomes significantly more challenging

for adversaries to deduce the original user queries, reconstruct sensitive entries from the knowledge base, or craft trigger-specific documents, thus mitigating model extraction and backdoor risks [16].

Knowledge Base Sanitization Regular sanitization of the knowledge base can prevent the ingestion of malicious documents. Techniques such as semantic consistency checks and provenance verification can detect and remove poisoned entries, including those designed for trigger-based attacks like *Phantom*. For instance, cross-referencing documents against trusted sources or flagging outliers in embedding space could reduce the risk of backdoor activation [21].

Privacy Protection Frameworks Given that RAG systems often process sensitive user data and rely on external knowledge bases, robust privacy protection is essential. Two promising approaches include:

Federated Learning In scenarios involving multiple institutions, federated learning allows each entity to train local models on their own data. This decentralized approach ensures that sensitive data remains on-site while contributing to a global model update, thereby reducing the risk of large-scale data breaches [17]. It also limits exposure to privacy exploits from trigger-based exfiltration attempts [21].

Anonymization and Data Sanitization Implementing techniques such as entity replacement, masking, and structured data sanitization can protect sensitive information both in user queries and in the retrieved documents. These methods ensure that confidential details (e.g., personal identifiers or medical records) are obfuscated before being processed or stored, thus

TABLE II
SUMMARY OF DEFENSE TECHNIQUES FOR RAG SYSTEMS

Defense Technique	Description	Advantages	References
Adversarial Training	Training with adversarial examples to improve robustness	Enhanced resistance to input manipulation	[13]
Dynamic Detection	Real-time monitoring for output anomalies	Timely attack detection	[14]
Model Watermarking	Embedding markers in generated text to track usage	Detection of model extraction	[15]
Differential Privacy	Adding noise to retrieval process to protect privacy	Privacy preservation	[16]
Knowledge Base Sanitization	Regular checks to remove malicious entries	Prevention of data poisoning	[21]
Class-RAG	Real-time content moderation using RAG	Rapid policy updates and robustness	[25]
Federated Learning	Decentralized training to protect local data	Reduced risk of data breaches	[17]
Anonymization	Removing identifiable information from data	Compliance with privacy regulations	[18]

complying with privacy regulations such as GDPR and HIPAA [18] and thwarting passage exfiltration objectives [21].

Summary Integrating these defense mechanisms creates a robust security architecture for RAG systems. Adversarial training and dynamic detection mechanisms help counteract input manipulations and trigger-based attacks, while model watermarking, differential privacy, and knowledge base sanitization safeguard the data and model layers. Moreover, federated learning and anonymization techniques provide comprehensive privacy protection. Together, these strategies ensure that RAG systems can be deployed securely even in high-stakes and sensitive environments.

V. FUTURE CHALLENGES AND RESEARCH DIRECTIONS

As RAG systems continue to evolve and integrate into critical applications, emerging challenges demand new research efforts to bolster their security, robustness, and transparency. This section outlines promising future directions, including dynamic attack defense, multi-modal threat handling, enhanced interpretability, and trigger-specific countermeasures, to guide the development of next-generation RAG systems.

Dynamic Attack Defense The open and adaptive nature of RAG systems renders them susceptible to dynamic adversarial attacks, where attackers continuously evolve their strategies to evade static defenses. A promising research direction is to model the interaction between adaptive attackers and defenders as a zero-sum game, employing game-theoretic frameworks [19]. In such models, the attacker's objective to maximize disruption is counterbalanced by the defender's goal to minimize potential loss, creating an iterative process of strategy refinement. This approach not only provides theoretical insights into the equilibrium of adversarial interactions but also lays the groundwork for automated, real-time defense mechanisms capable of dynamically adjusting to evolving threats, including trigger-based attacks [21].

Multi-modal Threat Handling The expansion of RAG systems into multi-modal domains, such as image-text fusion,

introduces additional security vulnerabilities. For example, in a multi-modal RAG system, adversaries might manipulate input images—through subtle tampering or replacement—to induce erroneous or biased text generation. Such image tampering could lead to significant deviations in the system's output, adversely affecting applications like medical diagnosis or security surveillance [20]. Future research should focus on developing robust multi-modal verification techniques, such as cross-modal consistency checks and anomaly detection algorithms, that can detect and mitigate these threats while maintaining the integrity of both visual and textual inputs.

Enhancing Interpretability Improving the interpretability of RAG systems is crucial for diagnosing security vulnerabilities and building user trust. One promising approach involves the use of SHapley Additive exPlanations (SHAP) to analyze the dependency of generated outputs on retrieved documents. By quantifying the contribution of each document to the final output, SHAP-based analysis can reveal hidden dependencies, potential biases, and trigger-induced anomalies in the retrieval process [22]. Enhanced interpretability not only facilitates debugging and system refinement but also provides critical insights for regulatory compliance and transparent decision-making in sensitive applications.

Countering Trigger-Based Attacks The *Phantom* attack [21] highlights the potency of trigger-based backdoor poisoning, where a single malicious document can manipulate outputs across diverse queries containing a specific trigger. Future research should explore specialized countermeasures, such as: - **Trigger Detection**: Developing algorithms to identify unusual retrieval patterns linked to specific tokens or phrases, potentially using statistical analysis of query-document similarity scores. - **Embedding Space Hardening**: Modifying retrieval embeddings to reduce sensitivity to trigger-specific alignments, possibly through regularization techniques or adversarial retraining. - **Behavioral Analysis**: Monitoring generator outputs for sudden shifts in tone, intent, or content (e.g., refusal, bias, or harm) that correlate

with retrieved documents, enabling real-time mitigation.

Additional Research Directions Beyond the aforementioned areas, several additional challenges merit further investigation:

1) **Scalable Defense Mechanisms:**

Developing lightweight yet effective defense strategies that scale with large datasets and high query volumes remains a significant challenge. Research into scalable anomaly detection and adaptive filtering techniques could yield robust, real-time defenses without incurring prohibitive computational costs.

2) **Cross-Domain Adaptation:**

RAG systems are deployed across diverse domains such as healthcare, finance, and legal services, each with unique security requirements. Future work should focus on creating adaptable defense frameworks that can be tuned to the specific vulnerabilities and regulatory landscapes of different domains.

3) **User-Centric Security:**

Integrating user feedback into security frameworks can enhance the adaptability of RAG systems. Investigating methods for incorporating user-driven security signals into automated defense mechanisms could lead to more resilient systems that evolve in response to real-world usage patterns.

Addressing these challenges will require a multidisciplinary approach that leverages advances in adversarial machine learning, game theory, and explainable AI. Through such efforts, the next generation of RAG systems can achieve a higher level of security, robustness, and transparency, ensuring their safe deployment in critical and high-stakes environments.

VI. CONCLUSION

This paper has provided a comprehensive, full-chain analysis of the security challenges in Retrieval-Augmented Generation (RAG) systems—from architectural vulnerabilities to sophisticated adversarial attacks, data poisoning, trigger-based backdoor attacks, and privacy leaks—and reviewed state-of-the-art defense techniques. The discussed defense strategies, including adversarial training, dynamic detection, model watermarking, differential privacy, knowledge base sanitization, and federated learning, underscore the complexity of securing RAG systems against evolving threats.

The security of RAG systems is directly related to their trustworthy application in sensitive domains such as healthcare, finance, and legal services. In these critical areas, any compromise in system integrity could result in significant adverse consequences, ranging from misinformation dissemination to breaches of sensitive personal data. Consequently, ensuring the security of RAG systems is not merely a technical challenge but a prerequisite for their safe and reliable deployment in real-world applications.

Moreover, the future challenges identified—such as dynamic attack defense modeled through game theory, multi-modal threat handling, enhanced interpretability using methods like SHAP, and countermeasures against trigger-based attacks—highlight the urgent need for continued research.

Addressing these challenges will require close collaboration between academia and industry, fostering innovative solutions that bridge theoretical advances and practical implementations.

Only through such interdisciplinary and collaborative efforts can robust, adaptive, and transparent RAG systems be developed to meet the stringent security demands of high-stakes applications.

REFERENCES

- [1] LEWIS, P., PEREZ, E., PIKTUS, A., PETRONI, F., KARPUKHIN, V., GOYAL, N., KÜTTLE, H., LEWIS, M., YIH, W.-t., ROCKTÄSCHEL, T., RIEDEL, S., KIELA, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.
- [2] LI, Y., LIU, G., WANG, C., YANG, Y. (2024). Generating is believing: Membership inference attacks against retrieval-augmented generation. *arXiv preprint arXiv:2406.19234*.
- [3] MOHAPATRA, P., RAKOVSKI, C. (2024). AI chatbots and challenges of HIPAA compliance for AI developers and vendors. *Journal of Medical Internet Research*, 26, e54870.
- [4] ZHOU, Y., WANG, Y., WANG, H., LIU, Z. (2024). Trustworthiness in retrieval-augmented generation systems: A survey. *arXiv preprint arXiv:2409.10102*.
- [5] ZOU, W., GENG, R., WANG, B., JIA, J. (2024). PoisonedRAG: Knowledge poisoning attacks to retrieval-augmented generation of large language models. *arXiv preprint arXiv:2402.07867*.
- [6] WU, Y., TANG, R., LIN, J. (2021). Adversarial attacks on neural retrieval models. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 1412–1421).
- [7] ZENG, S., ZHANG, J., HE, P., XING, Y., LIU, Y., XU, H., REN, J., WANG, S., YIN, D., CHANG, Y., et al. (2024). The good and the bad: Exploring privacy issues in retrieval-augmented generation (RAG). In *Findings of the Association for Computational Linguistics: ACL 2024*.
- [8] WAN, A., CHEN, Y., ZHANG, T., WANG, S., LIU, Q. (2023). Poisoning language models during instruction tuning. *arXiv preprint arXiv:2305.00944*.
- [9] JIANG, Y., ZHANG, S., WANG, X., LI, X., ZHANG, Y. (2024). Composite backdoor attacks against large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*.
- [10] EBRAHIMI, J., RAO, A., LOWD, D., DOU, D. (2018). HotFlip: White-Box Adversarial Examples for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 31–36).
- [11] TRAMÈR, F., ZHANG, F., JUELS, A., REITER, M. K., RUBIN, A. D. (2016). Stealing machine learning models via prediction APIs. In *Proceedings of the 25th USENIX Security Symposium* (pp. 601–618).
- [12] European Union. (2016). General Data Protection Regulation (GDPR). Regulation (EU) 2016/679.
- [13] GOODFELLOW, I. J., SHLENS, J., SZEGEDY, C. (2015). Explaining and harnessing adversarial examples. In *Proceedings of the 3rd International Conference on Learning Representations*.
- [14] BANG, K., LEE, H., LEE, J. (2022). Detection of adversarial examples in text classification: Benchmark and baseline via robust density estimation. In *Findings of the Association for Computational Linguistics: ACL 2022* (pp. 289–300).
- [15] KIRCHENBAUER, J., SCHWARZSCHILD, A., JAIN, N., WEN, Y., SOMEPELLI, G., CHIANG, P.-y., GOLDBLUM, M., SAHA, A., GEIPING, J., GOLDSTEIN, T. (2023). A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning* (pp. 17061–17084).
- [16] YANG, G., ZHANG, S. (2018). Differential privacy for information retrieval. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining* (pp. 662–670).
- [17] MCMAHAN, H. B., MOORE, E., RAMAGE, D., HAMPSON, S., ARCAS, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* (pp. 1273–1282).
- [18] SWEENEY, L. (2002). k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5), 557–570.

- [19] DASGUPTA, P., COLLINS, J. B. (2019). A survey of game theoretic approaches for adversarial machine learning in cybersecurity tasks. *AI Magazine*, 40(4), 31–43.
- [20] DOU, Y., LI, X., ZHANG, Y., LI, H. (2024). Adversarial attacks to multi-modal models. *arXiv preprint arXiv:2409.06793*.
- [21] CHAUDHARI, H., CHEN, P., DING, Y., JU, T., WU, Z., DU, W., YI, P., ZHANG, Z., LIU, G. (2024). Phantom: General Trigger Attacks on Retrieval Augmented Language Generation. *arXiv preprint*.
- [22] LUNDBERG, S. M., LEE, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (pp. 4768–4777).
- [23] CHENG, P., DING, Y., JU, T., WU, Z., DU, W., YI, P., ZHANG, Z., LIU, G. (2024). TrojanRAG: Retrieval-Augmented Generation Can Be Backdoor Driver in Large Language Models. *arXiv preprint arXiv:2405.13401*.
- [24] ASAI, A., WU, Z., WANG, Y., SIL, A., HAJISHIRZI, H. (2023). SELF-RAG: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.
- [25] CHEN, J., SHEN, E., BAVALATTI, T., LIN, X., WANG, Y., HU, S., SUBRAMANYAM, H., VEPURI, K. S., JIANG, M., QI, J., CHEN, L., JIANG, N., JAIN, A. (2024). Class-RAG: Real-Time Content Moderation with Retrieval Augmented Generation. *arXiv preprint arXiv:2410.14881*.