AI 기반 디지털 포렌식 자동 분석 시스템

권승원[1]*, 강지혁[2], 고은이[3], 최경규[4], 이병천(지도교수)[5]
중부대학교[1]

AI-Based Digital Forensics Automatic Analysis System

Seungwon Kwon[1]*, Jihyuck Kang[2], Euni Go[3], Gyeonggyu Choi[4], ByeongChun Lee[5]

요약: 본 연구는 LLM과 RAG을 통합하여 디지털포렌식 자동화 분석 시스템을 제안한다. MCP을 통해 LLM이 포렌식 도구를 제어하여 E01에서 아티팩트를 추출하고, HyDE 기반 선별 검색과 Reranking을 통해 VDB에서 관련 증거를 효율적으로 검색한다. 실험 결과, 제안 시스템은 침해사고 관련 질의에 대해 주요 공격 행위를 근거 기반으로 추론할 수 있었으며, 자연어 기반 포렌식 분석 자동화의 가능성을 확인하였다.

Key Words : Digital Forensics, Large Language Model, Retrieval-Augmented Generation, Automated Evidence Analysis, Model Context Protocol

1. 서론

사이버 공격이 고도화됨에 따라 디지털포렌식은 침해 원인 분석과 위협 인텔리전스 확보를 위한 핵심 기술로 자리 잡고 있다. 그러나 포렌식 과정에서 수집되는 E01 디스크 이미지는 방대한 비정형 데이터를 포함하며, 기존 상용 도구들은 수동 또는 반자동 방식으로 데이터를 처리하기 때문에 대규모 환경에서의 효율적 분석이 어렵다[1]. 이러한 한계를 극복하기 위해 본 연구는 대규모 언어 모델과 검색 증강 생성을 결합한 포렌식 분석 자동화 시스템을 제안한다.

2. 기술 배경 및 관련 연구

2.1 대규모 언어 모델(LLM)과 RAG

최근 대규모 언어 모델(LLM)을 활용한 포렌식 자동화 연구가 활발히 이루어지고 있다. Loumachi et al.[2]은 LLM과 규칙 기반 AI를 결합한 GenDFIR 프레임워크를 통해 로그 기반 사건 타임라인을 자동 생성하였다. 그러나 LLM은 hallucination 문제로 실제 증거와 무관한 정보를 생성할 수 있어 결과의 신뢰성에 한계가 있다. 또한 검색 증강 생성 (Retrieval-Augmented Generation, RAG) 기법은 검색 노이즈로 인해 관련성이 낮은 결과를 반환할 수 있다는 지적이 있다.[3] 이러한 문제를 보완하기 위해 HyDE(Hypothetical Document Embeddings) 및 Reranking 기법을 활용하여, 검색 정확도와 응답의 신뢰성을 높이는 방안을 제시한다.

2.2 Model Context Protocol (MCP)

MCP는 LLM과 외부 도구 간 표준 통신 프로토콜로, 복잡한 도구 체인 실행과 세션 상태 관리를 지원한다. 최근 연구에서 MCP를 LLM이 외부 도구 및 데이터 소스와 상호작용하도록 지원하는 개방형 표준으로 정의하며, 포렌식 환경에서 투명성과 재현성을 향상시킬 수 있음을 제시하였다[4]. 본 연구에서는 MCP를 통해 포렌식 도구를 등록하여 활용하고, LLM이 자연어 질의에 따라 적절한 도구를 호출하도록 설계하였다.

3. 시스템 설계 및 구현

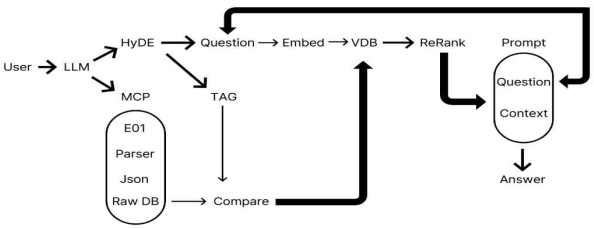

Fig. 1. 시스템 구성도

3.1 시스템 구현 환경 및 모델 구성

본 시스템은 그림 1과 같이, MCP가 포렌식 도구를 제어해 E01 이미지로부터 아티팩트를 추출하고 전처리하는 증거 추출 및 전처리 과정, HyDE가 질의를 확장하고 핵심 태그(TAG)를 생성해 벡터 데이터베이스(Milvus)에 색인하는 지능형 검색 및 색인 과정(임베딩 모델: text-embedding-paraphrase-multilingual-minilm-v2.gguf), 검색 결과를 Reranking → Prompt → Answer 순으로 처리하는 RAG 및 검증 과정(LLM: Qwen 3 4B Thinking-2507, Reranking 모델: bge-reranker-v2-m3-Q5_K_M-GGUF)으로 구성된다.

3.2 증거 추출 및 전처리

MCP 서버의 환경을 기반으로 LLM이 포렌식 도구를 직접 제어할 수 있도록 설계되었다. LLM은 MCP 서버를 통해 ArtifactExtractor를 호출하여 포렌식 이미지(E01)에서 아티팩트를 추출하고, Eric Zimmerman's Tools를 자동 실행하여 JSON 형태로 변환한다. 이 과정에서 MCP는 각 도구의 실행 결과를 LLM이 해석이 가능한 형태로 반환하며, 각 레코드에는 원본 증거와의 연계성을 확보하기 위한 메타정보(타임스탬프, 파일 경로, 레코드 유형, 해시값 등)가 자동으로 태깅한다. 변환된 데이터는 PostgreSQL 데이터베이스에 저장되며, 사건 유형, 관련 시스템, 사용자 계정 등 다양한 속성 기반 태그로 분류된다. MCP 서버는 RAG 검색 단계에서도 데이터베이스 질의를 중개하여, LLM이 자연어 질문을 SQL로 변환하고 관련 증거를 검색할 수 있도록 지원한다. 이를 통해 분석가는 복잡한 명령어 없이 대화형 인터페이스만으로 증거 추출부터 분석까지 일관된 워크플로우를 수행할 수 있다.

3.3 지능형 검색 및 색인

RAG 시스템의 검색 효율성을 높이고 Tag 구분을 명확히 하여 최종 답변 정확도를 향상시키는 핵심 단계로, 사용자 질문이 입력되면 HyDE 기반 LLM이 해당 질문에 대한 가상의 답변을 생성하고, 필수적인 증거 Tag를 정제한다. 이후 정제된 Tag와 연관된 증거 데이터만을 선택하여 벡터 임베딩을 수행한 뒤 VDB에 저장함으로써, 노이즈 데이터를 사전에 필터링하고 검색 공간을 크게 축소한다. 이러한 과정은 RAG의 초기 검색 단계에서 노이즈를 최소화하고, LLM이 보다 신뢰성 있고 관련성 높은 Context를 기반으로 사실 기반 답변을 생성할 수 있도록 지원한다.

3.4 RAG 및 검증

RAG 검색 단계에서는 사용자의 질문 벡터와 VDB에 색인 된 증거 벡터 간 유사도를 계산하여 상위 K개의 Context를 추출하며, 이후 Reranking 단계에서 크로스-인코더 기반 Reranking 모델을 통해 각 Contexts가 질문과 얼마나 관련성이 높은지 재평가하고 순위를 재조정한다. 최종적으로 가장 관련성이 높은 Contexts만이 메인 LLM에 입력되어, 증거 기반으로 신뢰성 있는 답변을 생성하도록 지원함으로써, 단순 검색에 비해 정확도와 사실 근거의 신뢰성을 동시에 확보한다.

4. 실험 및 평가

4.1 실험 환경 및 데이터셋

실험은 실제 사용자 환경을 반영하기 위해 Windows 11 기반의 디스크 이미지를 활용하였다. 사용된 디스크 이미지는 다양한 침해 행위를 포함한 E01 포맷으로 구성되었으며, 주요 시나리오는 DGA(Domain Generation Algorithm) 기반 도메인 생성, 내부 정보 유출 시도, 백신 설정 변경 및 무력화, 서비스 등록을 통한 지속성 확보, USB HID 공격 시도 등을 포함한다.

4.2 평가 방법

평가는 사전에 정의된 질의(Q) 집합을 기반으로 수행하였다. 각 질의에 대해 시스템이 생성한 응답(A*)이 기준 답변(A)을 직접 포함하거나, A*의 내용으로부터 A를 논리적으로 도출할 수 있는지를 검증하였다. 즉, LLM이 생성한 서술을 통해 정답과 동일한 결론에 도달할 수 있는지를 판단함으로써, 시스템의 추론 정확성과 근거 일관성을 평가하였다.

4.3 실험 결과

table. 1 포렌식 분석 범주별 정확도 결과

항목	질의 수	정확(1)합	부분(0.5)합	정확도(%)
Activity	5	2	2	60
Intrusion	5	3	1	70
Exfiltration	5	0	1	10

Table 1은 제안된 시스템의 응답 정확도를 항목별로 제시한다. 행위분석(Activity) 범주에서는 USB 연결 기록과 사용자 활동 로그가 일관되게 식별되어 정확도 60%를 보였으며, 시스템이 정상 행위 기반의 이벤트를 효과적으로 구분함을 확인할 수 있다. 보안 우회(Defense Evasion) 범주에서는 Windows Defender 설정 변경(Event ID 5007)과 서비스 등록 (Event ID 7045) 등 보안 정책 변조 및 지속성 확보 행위가 탐지되어 정확도 70%를 기록하였고, 시스템이 내부 침해 과정에서 발생하는 보안 회피 패턴을 적절히 포착함을 보여준다. 반면 유출판별(Exfiltration) 범주에서는 PowerShell 실행과 압축 파일 생성이 관찰되었으나, 실행 주체와 후속 네트워크 활동 간의 연계성이 부족하여 행위의 의도를 명확히 판단하기 어려웠으며, 정확도는 10%로 낮게 나타났다. 이러한 결과는 제안된 시스템이 실제 로그를 근거로 행위를 일정 수준에서 구분하고 침해 징후를 부분적으로 추론할 수 있음을 시사한다.

5. 결론

본 연구는 대규모 언어 모델과 RAG 아키텍처를 결합하여 포렌식 증거 분석을 자동화하는 시스템을 제안하였다. HyDE 기반 증거 선별과 Reranking 기법을 통해 방대한 E01 디스크 이미지에서도 관련성 높은 증거를 효율적으로 검색하고 사실 기반의 응답을 생성할 수 있음을 확인하였다. 다만, 벡터 임베딩 처리 속도 지연과 단일 디스크 이미지에 의존한 실험 설계로 인해 행위 간 인과관계 추론에는 한계가 존재하였다. 향후에는 네트워크 패킷, 메모리 덤프 등 다양한 증거 소스를 통합하고, 증분 임베딩 및 시간적 상관관계 분석 모듈을 적용함으로써, 보다 더 정교한 공격 연쇄 분석과 자동화된 포렌식 추론 체계의 고도화가 가능할 것으로 기대된다.

참고문헌

[1] T. Mohammed, F. K. H. Al-Azawey, and A. T. S. Al-Sultani, "An Automated Approach for Digital Forensic Analysis of Heterogeneous Big Data," *Int. J. Comput. Appl. *, vol. 145, no. 12, pp. 28-33, 2016.

[2] Loumachi, A., et al., *GenDFIR: Leveraging Large Language Models and Rule-Based AI for Digital Forensics Automation*, Forensic Science International: Digital Investigation, 2023.

[3] Gao, T., Yao, X., and Chen, D., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *arXiv preprint arXiv: 2202.08904*, 2022.

[4] J.-N. Hilgert, C. Jakobs, M. Külper, M. Lambertz, A. Mahr, and E. Padilla, “Chances and Challenges of the Model Context Protocol in Digital Forensics and Incident Response,” arXiv:2506.00274, 2025.