Journal of Convergence for Information Technology
Vol. 11. No. 6, pp. 24-32, 2021
e-ISSN 2586-4440
DOI : https://doi.org/10.22156/CS4SMB.2021.11.06.024

랜섬웨어 탐지를 위한 그래프 데이터베이스 설계 및 구현

최도현
숭실대학교 컴퓨터학과 학생

Graph Database Design and Implementation for Ransomware Detection

Do-Hyeon Choi
Student, Dept. of Computer Science, Soongsil University

요 약 최근 랜섬웨어(ransomware) 공격은 이메일, 피싱(phishing), 디바이스(Device) 해킹 등 다양한 경로를 통해 감염되어 피해 규모가 급증하는 추세이다. 그러나 기존 알려진 악성코드(정적/동적) 분석 엔진은 APT(Aadvanced Persistent Threat)공격처럼 발전된 신종 랜섬웨어에 대한 탐지/차단이 매우 어렵다. 본 연구는 그래프 데이터베이스를 기반으로 랜섬웨어 악성 행위를 모델링(Modeling)하고 랜섬웨어에 대한 새로운 다중 복합 악성 행위를 탐지하는 방법을 제안한다. 연구 결과 기존 관계형 데이터베이스와 다른 새로운 그래프 데이터베이스 환경에서 랜섬웨어의 패턴 탐지가 가능함을 확인하였다. 또한, 그래프 이론의 연관 관계 분석 기법이 랜섬웨어 분석 성능에 크게 효율적임을 증명하였다.

주제어 : 악성코드, 그래프 데이터베이스, 랜섬웨어, 행위 분석, 연관 분석

Abstract Recently, ransomware attacks have been infected through various channels such as e-mail, phishing, and device hacking, and the extent of the damage is increasing rapidly. However, existing known malware (static/dynamic) analysis engines are very difficult to detect/block against novel ransomware that has evolved like Advanced Persistent Threat (APT) attacks. This work proposes a method for modeling ransomware malicious behavior based on graph databases and detecting novel multi-complex malicious behavior for ransomware. Studies confirm that pattern detection of ransomware is possible in novel graph database environments that differ from existing relational databases. Furthermore, we prove that the associative analysis technique of graph theory is significantly efficient for ransomware analysis performance.

Key Words : Malware, Grape Database, Randomware, Behavior Analysis, Association Analysis

1. 서론

최근 랜섬웨어 공격은 악성코드의 한 유형으로써 개인에서 기업과 기관으로부터 비트코인을 요구하는 등의 악성코드 유형이다. 최근 수년 동안 악성코드는 의료, 생산 및 제조업, 상품 추천, 무인 자동차와 드론 등 위협 대상이 전 분야로 확대되고 있다[1]. 이에 따라, 보안 분야는 학습 가능한 데이터셋(Data Set)을 기반으로 악성코드를 탐색하고 위협을 예측하는 기술에 관한 연구에 관심이 높아지고 있다. “2018~2019 정보보호 R&D 데이터 챌린지”는 AI를 기반으로 악성코드, 주요 취약점, 게임봇(Bot), 자동차용 침입 등 탐지 엔진의 탐지율 80% 이상 목표로 악성코드를 분석했다[2]. 보안 분야의 주요 기술 발전 방향이 기존 누적된 위협 관

*Corresponding Author : Do-Hyeon Choi(cdhgod0@ssu.ac.kr)
Received March 25, 2021
Revised May 24, 2021
Accepted June 20, 2021
Published June 28, 2021

련 빅데이터를 어떻게 분석하는가에 집중되고 있다. 앞으로 데이터 분석 분야는 고도화 단계에서 신규/변종 악성코드 탐지와 전문가 인력의 부족 등 보안 분야의 다양한 문제점을 해결할 수 있을 것으로 기대되고 있다.

본 연구는 NOSQL 계열(비정형 데이터베이스)의 그래프 데이터베이스(GDB : Graph database) 기반으로 랜섬웨어 탐지 모델을 설계하고 개발한다. 그래프 이론을 기반으로 랜섬웨어 악성 행위를 새로 모델링 하였고, 기존 정적/동적 분석 엔진과는 다른 그래프 기반 악성코드 탐지 방법을 제안하였다. 본 논문의 구성은 다음과 같다. 2장 관련 연구에서 기존 랜섬웨어 연구 비교분석, 3장 랜섬웨어 그래프 모델링 및 탐지, 4장은 모델 검증 및 성능 분석, 5장 결론으로 마친다.

2. 관련 연구

2.1 기존 관계형 데이터베이스의 한계

관계형 데이터베이스(RDB : Relational database)는 가장 널리 사용되고 있는 데이터베이스이다. 수년 사이 클라우드 플랫폼의 고도화와 함께 빅데이터 활용의 중요도가 높아졌고, 데이터베이스 처리 기술이 크게 변화하고 있다. 특히, 비정형 데이터를 대상으로 의미 있는 데이터를 분석하는 AI 분야에 대한 기대가 높다. 기존 정형화된 구조로 설계된 RDB는 비정형/반정형 등 실시간 빅데이터를 저장 및 처리에 한계가 존재한다. Fig. 1은 기존 SQL을 포함하는 정형, 이외 반정형/비정형 데이터를 나타낸다[3]. 이메일 또는 웹페이지, 소스코드 파일 등 반정형 데이터와 이미지, 오디오, 영상 등 비정형 데이터는 특정 정형화된 구조가 존재하지 않는다. Venkatraman에 의하면 RDB 계열의 SQL와 비교하여 NOSQL 데이터베이스가 85% 이상 성능이 뛰어남을 증명했다[4].

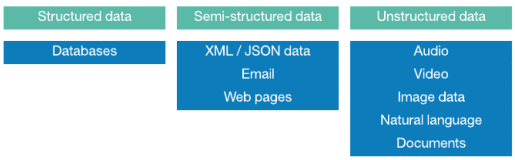

Fig. 1. Examples of Structured, Semi-structured and Unstructured Data

김기성에 의하면 전용 벤치마크(YCSB)에서 처리속도 및 지연 등 항목 대부분에서 NOSQL 데이터베이스가 성능이 뛰어남을 증명했다[5]. Fig. 2는 RDB와 NOSQL 계열의 데이터베이스의 성능 비교 분석 결과를 나타낸다. 기존 SQL과 NOSQL을 같이 활용하는 멀티 데이터베이스 모델의 예이다[6].

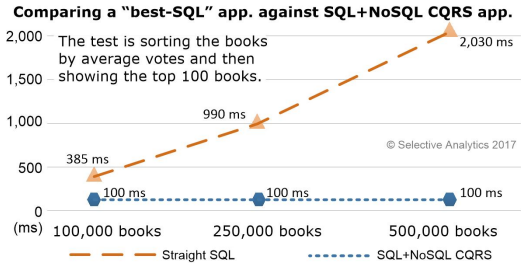

Fig. 2. SQL vs NOSQL Performance Analysis Results(Avg) - Book Sample Example

평균 투표 값을 계산하고 정렬 작업 수행 결과 기존 SQL은 25만 서적 구간을 넘어가면서 지연 시간(ms)이 급격하게 증가(9배 이상)했다. 지연 시간이 증가하면 성능이 크게 떨어진다. NOSQL 계열 데이터베이스는 AI 플랫폼 및 성능 효율성 등 장점이 많지만 해결해야 할 문제가 있다. 비정형 데이터는 말 그대로 기존 문자 조합이 아닌 모든 데이터를 대상으로 분석하기 때문에 정보의 수집과 처리하기 위한 1차 데이터 전처리 과정이 매우 어렵다[7]. 랜섬웨어와 같은 악성코드 유형의 데이터 전처리를 고려해야 한다. 기존 잘 구축된 악성코드 RDB와 NOSQL 계열을 병행하는 멀티 데이터베이스 모델이 이를 해결해 줄 것이다. NOSQL 전용 데이터베이스로 완전히 전환되려면 비정형 데이터 누적과 처리 엔진 기술이 고도화되어야 한다. 이는 아직 보안 분야에서 기술 초기 단계로써 관련 새로운 연구에 더 많은 시간을 투자해야 한다는 것을 의미한다.

2.2 기존 랜섬웨어 탐지/분석 연구분석

최근 수년 내 랜섬웨어 관련 연구들은 기존 정적/동적 엔진 분석의 한계를 극복하기 위한 탐지 개선 연구와 기계학습과 딥러닝 기법(정적분석)을 적용한 연구로 분류된다. Table 1은 최근 수년 내 랜섬웨어 관련 대표 연구를 나타낸다[8-14].

Table 1. Ransomware-related Research Issues

Author	Proposed	Problem
Ko,Y.S.	Platform-independent structures and restore techniques	Lack of specific development measures, relying on restoration techniques
Lee, J. w.	Detections by placing bait files	Possible pattern bypass in the form of a decoy file
Park, S. K.	Kernel-level backup methods	Relies on proactive technology
Joo, J. G.	Machine learning-based feature extraction/analysis	Requires combination of machine learning algorithms from various perspectives other than SVMs
Kim, H. B.	Extract/analyze PE structural features	Limits of existing PE feature extraction ranges
Joe, W. J.	Behavior similarity-based variant detection	Limitations of traditional PE feature extraction, limit of 1000 sample data
Seong Il Bae	Machine learning-based feature extraction/analysis	Select various algorithms (high accuracy), limit system call area

최근 연구들을 살펴보면 대부분 랜섬웨어 탐지율과 랜섬웨어 우회 기법들을 개선하기 위한 연구이다. 비교분석 결과 미끼 회피 방식, 커널 레벨 백업 방식, 최적 특징 선택, 특징 정제 기술, 행위기반 유사도 등 구체적인 설계 및 모델에 대한 정의가 부족하고, 공격에 대한 탐지보다 우회 및 예방 연구임을 알 수 있다[8-10, 12, 13]. 기계학습 기반 연구는 앞으로 탐지 정확도를 크게 개선할 수 있을 것으로 기대되고 있다. 초기 연구 단계로써 SVM(support vector machine)과 Naive Bayes과 같은 기계학습 알고리즘부터 딥러닝 수준의 RNN 알고리즘을 선택하고 최적화를 진행하고 있다[11, 14-18]. 중요 문제는 연구분석에 사용한 악성코드 샘플(Sample)의 양이 부족하고, 실제 기존 RDB와 객관적인 성능을 비교할 만한 연구가 부족하다는 것이다. 악성코드 탐지를 위한 새로운 기법과 함께 앞서 관련 연구분석 결과와 같이 기존 RDB의 성능 한계 문제를 해결할 수 있어야 한다. 고성능 CPU와 GPU 등은 큰 비용 문제와 함께 근본적인 성능 최적화 문제를 해결할 수 없다. 이현종 외 2명은 악성코드를 효율적으로 저장과 처리를 가속화 할 수 있는 Hadoop 기반 분산 처리 시스템을 제안했다[19]. 시스템 외부 네트워크를 통해 다중파일을 조작하고 암호화하는 복합적인 행위 등을 분석(2만 개 샘플)하는데 성능이 평균 3.75배 향상되었다. 랜섬웨어와 같은 악성코드 처리에 비정형 데이터 분산 처리 시스템이 성능이 뛰어남을 증명하고 있다.

연구분석 결과, 악성코드 샘플 데이터로부터 PE와 API 그리고 시스템 콜 영역 등 제한적인 분석 범위를 확장할 필요가 있다. 또한, 악성코드 분석에 새로운 플랫폼과 전용 데이터베이스의 적용이 요구됨을 알 수 있다.

3. GDB 기반 랜섬웨어 탐지 설계 및 구현

3.1 랜섬웨어 다중 행위 데이터 정의

Table 2는 본 연구에서 사용한 랜섬웨어 행위 분석 세부 항목을 나타낸다. 본 연구는 기존 보유하고 있는 악성코드 RDB를 기반으로 분석 대상을 프로세스, PE 파일, 레지스트리, 네트워크 등 세부 범위를 확장하였다. 제안하는 새로운 GDB 스키마로 모델링을 위해 다양한 악성 행위 항목을 새롭게 정의하였다.

Table 2. Behavior Analysis Data Definition

Attribute	Type	Description
PC_ID	long	PC ID
C_Time	long	Collecting Time
Action	string	Suspicious Action
Process	Struct	Current Process
P_ID	long	Process ID
Pre_ID	long	Previous P_ID
Target_ID	long	Target P_ID
PE_File	Struct	PE File Information
F_Name	string	File Name
F_Path	string	File Path
F_Size	long	File Size
Registry	Struct	Registry Information
H_Type	string	Hive Type
K_Name	string	Key Name
V_Name, V_Data	string	Value Name, Data
Network	Struct	Network Information
L3_Protocol	string	L3 Protocol(IP)
L4_Protocol	string	L4 Protocol(TCP, UDP)
L7_Protocol	string	L7 Protocol(HTTP)
Source_IP	string	Source IP
Source_Port	integer	Source Port
Dest_IP	string	Destination IP
Dest_Port	integer	Destination Port
Send_Data	string	PayLoad

3.2 랜섬웨어 다중 행위 그래프 모델링

Fig. 3은 기존 행위탐지 구조(상단)를 그래프 모델링한 전체 연결 및 흐름을 나타낸다. 기존 행위탐지 구조는 악성코드 테이블을 전수조사하여 탐지한 후, 조인(Join) 연산을 반복하는 구조이다. 그래프 이론을 기반(하단)으로 현재 프로세스에 연결된 PE 파일의 생성과 실행, 자동등록, 삭제 등 악성 행위들을 연결하였다.

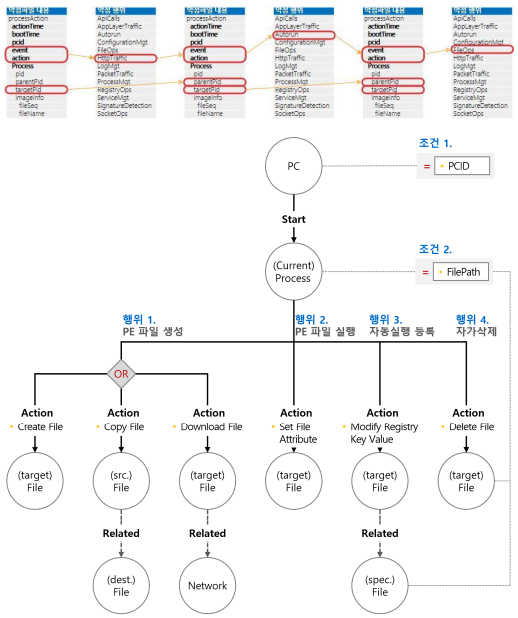

Fig. 3. Ransomware Malicious Behavior - Full Modeling

초기 프로세스로부터 파생된 PE 파일 생성과 실행과 같은 기본적인 2가지 악성 행위와 자동실행 되도록 레지스트리를 변조하는 행위, 특정 프로세스에 인잭션을 시도하는 행위에 대해 검사를 수행한다. Fig. 4-7은 주요 그래프 연결 관계와 세부 행위를 나타낸다. PC는 검사 대상을 나타낸다. 프로세스 이벤트를 시작으로 파일, 레지스트리, 네트워크 항목들을 연계하여 악성 행위를 탐지하는 형태이다.

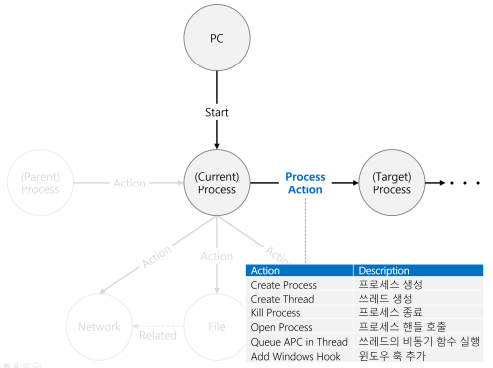

Fig. 4. Ransomware Malicious Behavior - Process Modeling

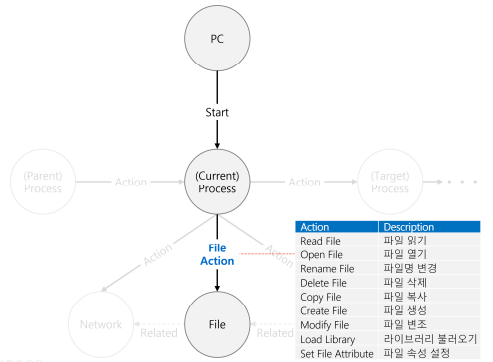

Fig. 5. Ransomware Malicious Behavior - File Modeling

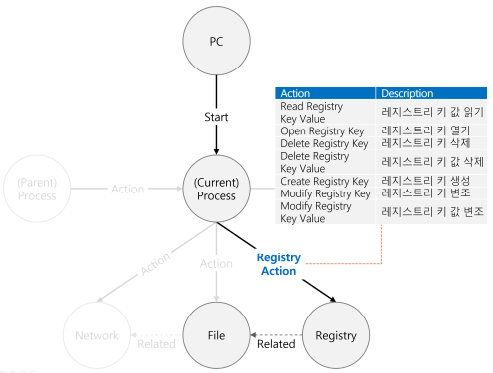

Fig. 6. Ransomware Malicious Behavior - Registry Modeling

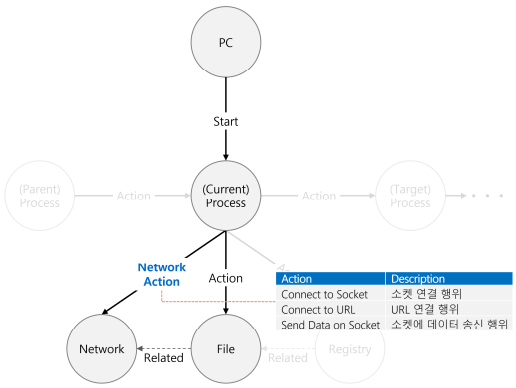

Fig. 7. Ransomware Malicious Behavior - Network Modeling

프로세스에서 생성(Create Thread)된 스레드에 독립적인 콜백 함수들을 모아둔 큐를 통해 필요할 때 호출(Queue APC in Thread)하거나, 윈도우 훅을 추가(Add Windows Hook)하는 행위 중 한 가지를 수행하여 인젝션 대기 상태를 만든다. 이후 프로세스, 레지스트리, 파일, 네트워크 등에 대한 행위를 검사한다.

4. 모델 검증 및 성능 분석

4.1 GDB 테스트 환경 설정

Table 3과 같이 실행 환경은 리눅스 기반의 Amazon S3를 사용하고, GDB를 적용했다. 원본 하둡 빅데이터를 전용 Parquet 파일 3.1Gb 파일로 필터링하고 변환(ETL)한 후 전용 CSV 파일로 추출했다. (ETL : Extract/Tranform/Load)

Table 3. Test Execution Environment Details

	Spec
OS	Amazon Linux AMI
CPU	30 CPU(Intel Xeon E5-2676 v3)
Memory	157GB
SSD	1TB

추출된 CSV 파일을 외부 테이블(External Table) 형태로 변환하고, 제안 그래프 모델에 따른 객체(Node)와 관계(Edge)를 생성한다. Fig. 8은 앞서 완성된 그래프 모델 및 데이터 적재 대시보드 화면 예이다. 프로세스 또는 네트워크와 파일 등은 모두 객체(Vertex)로 정의되며 앞서 모델링한 테이블 내 속성정보들을 포함한다. 화살표로 각 악성 행위 관계(Edge)를 연결되어 있음을 볼 수 있다.

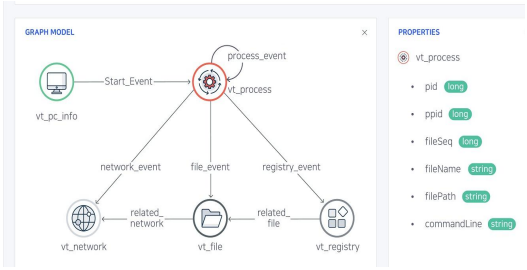

Fig. 8. Example of Graph Model and Data Load Dashboard Screen

4.2 GDB 기반 랜섬웨어 검증

GDB 기반으로 랜섬웨어(총 5가지 유형)의 파일 변조 및 삭제와 암호화 등 악성 행위를 사이퍼(Cyper) 쿼리로 Match 한다. Fig. 9는 다중파일의 이름을 변경한 뒤 파일 내용을 수정(암호화)하는 패턴을 탐지한 결과이다. 초록색 객체(vt_pc_Info)는 4대(2, 1, 1)를 시작으로 붉은색 객체(vt_process) 3개(1, 1, ,1) 이후 회색 객체(vt_file) 16(4, 4, 8)개 순서로 관계가 분산되어 연결되어 있음을 나타낸다.

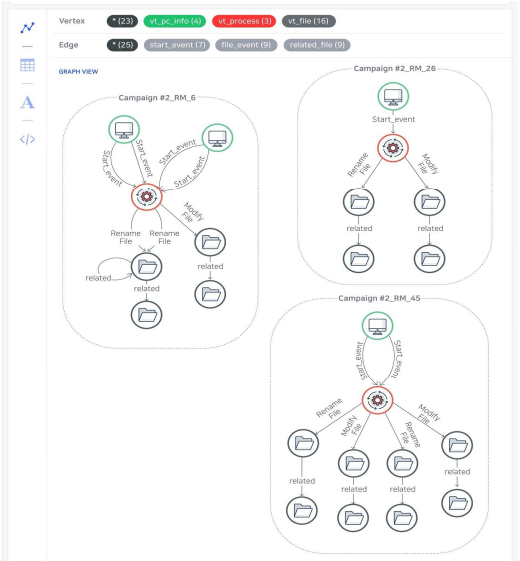

Fig. 9. Ransomware Pattern(Rename-Modify)

Fig. 10은 다중파일을 복사한 후 파일 내용을 수정

(암호화)하는 패턴을 탐지한 결과이다.

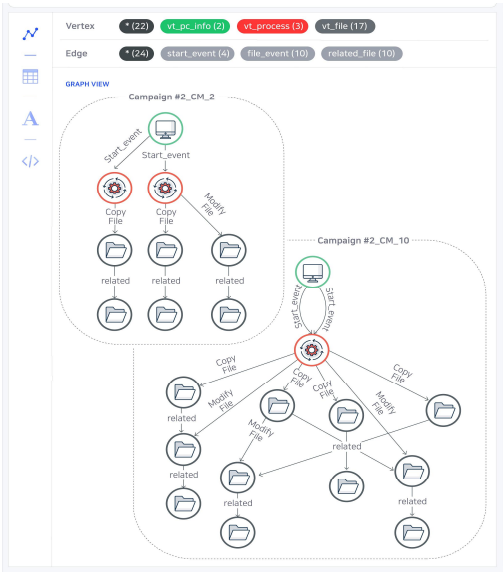

Fig. 10. Ransomware Pattern(Copy-Modify)

Fig. 11은 다중파일을 암호화하고, 해당 문서들을 삭제하는 패턴을 감지한 결과이다.

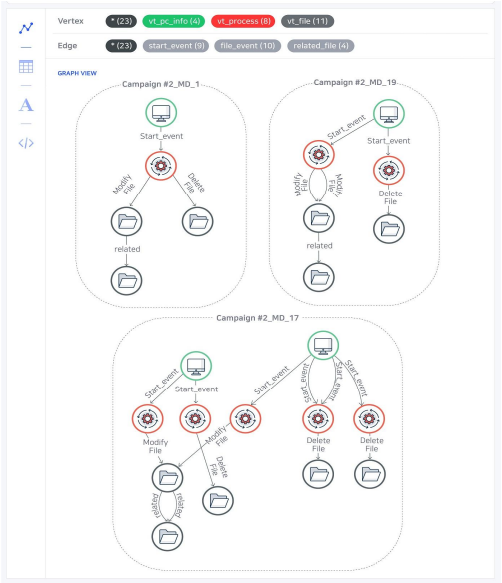

Fig. 11. Ransomware Pattern(Modify-Delete)

Fig. 12는 다중파일의 이름을 변경한 후, 해당 문서들을 암호화하는 패턴을 감지한 결과이다.

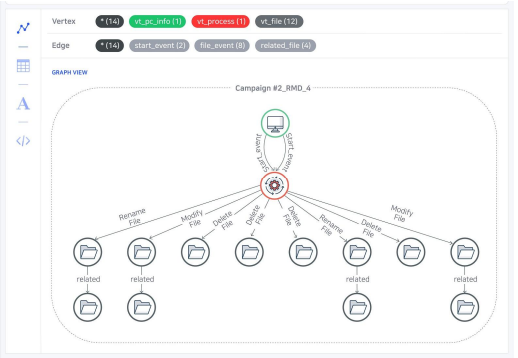

Fig. 12. Ransomware Pattern(Rename-Modify-Delete)

Fig. 13은 다중파일을 복사한 후 암호화하고, 해당 문서들을 삭제하는 패턴을 감지한 결과이다.

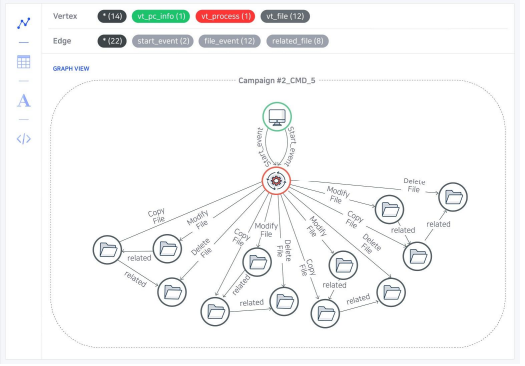

Fig. 13. Ransomware Pattern(Copy-Modify-Delete)

검증과정은 시작 객체(Start_event)로 이후 객체의 연결 관계를 1단계(depth)로 정의한다. 그래프 시각화 엔진을 통해 직관적으로 악성 행위에 대한 연관 관계를 확인하였다. 악성 패턴은 Match 결과 전체 3단계~5단계 수준에서 집중되어 나타났다.

4.3 RDB와 GDB 성능 분석

본 성능 분석과정은 백신 소프트웨어를 공급하는 A사 등에서 활용 중인 RDB 기반의 행위 탐지 모델을 대상으로 제안 GDB 기반 모델의 악성 행위 탐지 유무와 탐지에 걸리는 시간 비용(Seconds)을 상호 비교한다. 기존 RDB 기반의 데이터는 테이블 간 관계는 연결되어 있지 않기 때문에, 추가적인 조인(Join) 연산을 수행하였다. Fig. 14-18은 앞서 검증과정과 같은 랜섬웨어

악성 패턴의 성능 비교분석 결과(순서대로)를 나타낸다. GDB 모델이 RDB 모델과 비교하여 성능이 월등하게 높은 것으로 나타났다. (소수점 3자리 반올림) 초 단위가 낮을수록 성능이 높다.

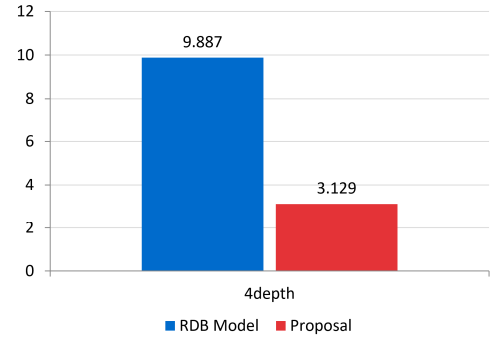

Fig. 14. Ransomware Detection Time Result(Rename-Modify)

4단계 이상의 악성 행위 Rename-Modify는 (RDB)9.887초에서 (제안)3.129초로 성능이 약 3.16배 (216%) 향상되었다.

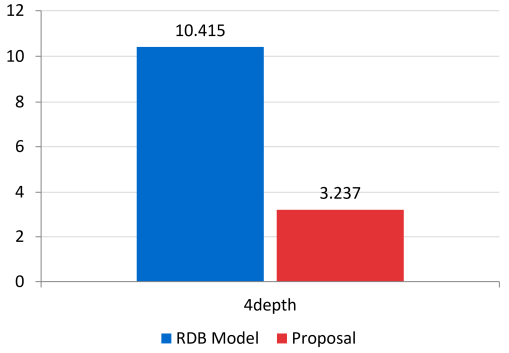

Fig. 15. Ransomware Detection Time Result(Copy-Modify)

4단계 이상의 악성 행위 Copy-Modify는 (RDB)10.415초에서 (제안)3.237초로 약 3.22배 (222%) 향상되었다.

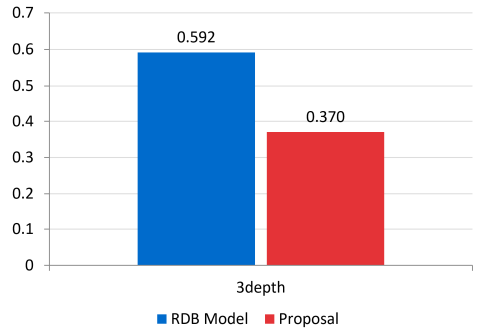

Fig. 16. Ransomware Detection Time Result(Modify-Delete)

3단계 수준에서 탐지된 Modify-Delete는 (RDB)0.592에서 (제안)0.392초로 성능이 약 1.60배 (60%) 향상되었다.

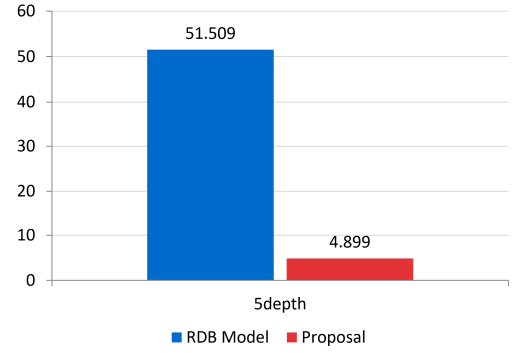

Fig. 17. Ransomware Detection Time Result(Rename-Modify-Delete)

5단계 수준에서 탐지된 Rename-Modify-Delete는 (RDB)51.509에서 (제안)4.899초로 성능이 약 10.51배(951%) 향상되었다.

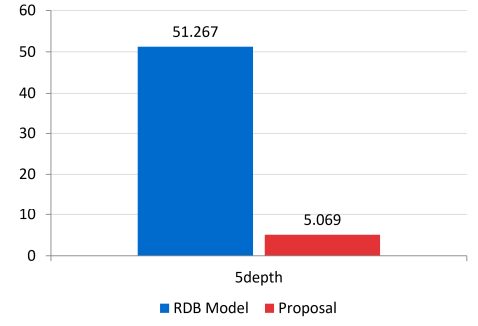

Fig. 18. Ransomware Detection Time Result(Copy-Modify-Delete)

5단계 수준에서 탐지된 Copy-Modify-Delete 행위

의 경우 (RDB)51.267에서 (제안)45.069로 성능이 10.11배(911%) 향상되었다. 전체 성능 분석 결과, 기존 RDB 기반의 탐지 엔진도 충분히 준수한 수준의 성능을 보여주지만, GDB가 악성 행위 단계가 증가할수록 성능이 크게 뛰어나다는 것을 확인했다.

5. 결론

본 연구는 기존 RDB 모델의 비효율적인 악성코드 분석 문제를 해결하고자, 다중행위를 중심으로 기존 RDB 모델을 확장하는 GDB 기반의 랜섬웨어 탐지 방법을 설계하고 구현하였다. 새롭게 생성된 그래프 모델은 랜섬웨어와 같은 악성코드 행위의 연관 관계를 분석하는데 효율적이라는 것을 확인하였다. 일반적인 랜섬웨어 패턴 5개를 중심으로 탐지 여부를 확인하였으며, 성능 분석 결과 전체 평균 성능이 RDB 모델보다 월등하다는 것을 확인하였다. 향후, 본 연구는 랜섬웨어에서 다중행위 패턴의 수준(단계)을 확장하여 악성코드 탐지/분석 범위를 넓히고 GDB와 같은 비정형 데이터 분석을 기반으로 기계학습 또는 딥러닝 알고리즘을 적용하는 연구로 발전시킬 계획이다.

REFERENCES

[1] S. H. Woo. (2020). Attack Types and Countermeasures of Next Generation Ransomeware. *Journal of the Korea Information and Communication Association Conference, 24(1),* 541-544.
UCI(KEPA) : I410-ECN-0101-2020-004-000905920

[2] S. J. Kim, J. H. Ha, S. H. Oh & T. J. Lee. (2019). A Study on Malware Identification System Using Static Analysis Based Machine Learning Technique. *Journal of the Korea Institute of Information Security & Cryptology, 29(4),* 775-784.
DOI : 10.13089/JKIISC.2019.29.4.775

[3] Arvind Padmanabhan. (Date of publication). Devopedia. *Structured vs Unstructured Data*(Online). https://devopedia.org/structured-vs-unstructured-data

[4] S. Venkatraman, K. Fahd, S. Kaspi & R. Venkatraman. (2016). SQL versus NoSQL movement with big data analytics. *Int. J. Inform. Technol. Comput. Sci., 8,* 59-66.
DOI : 10.5815/ijitcs.2016.12.07

[5] K. S. Kim. (2016). Performance Comparison of PostgreSQL and MongoDB using YCSB. *Journal of Korean Institute of Information Scientists and Engineers, 43(12),* 1385-1395.
UCI(KEPA) : I410-ECN-0101-2017-569-001860058

[6] Jon. P. Smith. (Date of publication). The Reformed Programmer. EF Core - Combining SQL and NoSQL databases for better performance. https://www.thereformedprogrammer.net/

[7] C. S. Bae & S. C. Goh. (2020). For Improving Security Log Big Data Analysis Efficiency, A Firewall Log Data Standard Format Proposed. *Journal of the Korea Institute of Information Security and Cryptology, 30(1),* 157-167.
DOI : 10.13089/JKIISC.2020.30.1.157

[8] Y. S. Ko & J. P. Park. (2019). A Study on the Ransomware Detection System Based on User Requirements Analysis for Data Restoration. *Journal of the Korea Academia-Industrial cooperation Society, 20(4),* 50-55.
DOI : 10.5762/KAIS.2019.20.4.50

[9] J. W. Lee, Y. M. Kim, J. H. Lee & J. M. Hong. (2019). An Efficient Decoy File Placement Method for Detecting Ransomware. *Journal of Korean Institute of Smart Media, 8(1),* 27-34.
DOI : 10.30693/SMJ.2019.8.1.27

[10] S. K. Park. (2020). Development of Prevention and Post-recovery System against the Ransomwares Attacks using the Technique of Massively Data Signing and Kernel Level Backup. *Journal of the Institute of Electronics and Information Engineers, 57(3),* 57-72.
DOI : 10.5573/ieie.2020.57.3.57

[11] J. G. Joo, I. S. Jung & S. H. Kang. (2019). An Optimal Feature Selection Method to Detect Malwares in Real Time Using Machine Learning. *Journal of Korea Multimedia Society, 22(2),* 203-209.
DOI : 10.9717/kmms.2019.22.2.203

[12] H. B. Kim & T. J. Lee. (2020). Stacked Autoencoder Based Malware Feature Refinement Technology Research. *Journal of the Korea Institute of Information Security & Cryptology, 30(4),* 593-603.
DOI : 10.13089/JKIISC.2020.30.4.593

[13] W. J. Joo & H. S. Kim. (2019). A Malware Variants Detection Method based on Behavior Similari. *Journal of Korean Institute of Smart Media, 8(4),* 25-32.
DOI : 10.30693/SMJ.2019.8.4.25

[14] S. I. Bae, G. B. Lee & E. G. Im. (2020).

Ransomware detection using machine learning algorithms. *Concurrency and Computation: Practice and Experience, 32(18)*.
DOI : 10.1002/cpe.5422

[15] J. H. Hwang & T. J. Lee. (2017). Android Malware Analysis Technology Research Based on Naive Bayes. *Journal of the Korea Institute of Information Security & Cryptology, 27(5),* 1087-1097.
DOI : 10.13089/JKIISC.2017.27.5.1087

[16] J. B. Yoo, S. J. Oh, R. H. Park & T. K. Kwon. (2018). Development Research of An Efficient Malware Classification System Using Hybrid Features And Machine Learning. *Journal of the Korea Institute of Information Security & Cryptology, 28(5),* 1161-1167.
DOI : 10.13089/JKIISC.2018.28.5.1161

[17] J. H. Ha & T. J. Lee. (2020). Research on text mining based malware analysis technology using string information. *Journal of Korea Internet Computing and Services, 21(1),* 45-55.
DOI : 10.7472/jksii.2020.21.1.45

[18] Y. B. Cho. (2018). The Malware Detection Using Deep Learning based R-CNN. *Journal of Korea Digital Contents Society, 19(6),* 1177-1183.
DOI : 10.9728/dcs.2018.19.6.1177

[19] H. J. Lee, S. Y. uh & D. S. wang. (2019). Distributed Processing System Design and Implementation for Feature Extraction from Large-Scale Malicious Code. *KIPS Transactions on Computer and Communication Systems, 8(2),* 2.
DOI : 10.3745/KTCCS.2019.8.2.35

최 도 현(Choi Do Hyeon) [정회원]

· 2008년 2월 : 동서울대학교 컴퓨터소프트웨어학과 졸업
· 2010년 8월 : 숭실대학교 컴퓨터학과(공학석사)
· 2016년 3월 : 숭실대학교 컴퓨터학과(공학박사)

· 관심분야 : Mobile, Network Security, PKI, Virtualization
· E-Mail : cdhgod0@ssu.ac.kr