[논문 주제 제안서]

[Windows 환경 기반 Discord 아티팩트 분석 방법론 비교 및 프레임워크 제안]

작성일	2025.05.26
작성자	정지윤, 전소현, 안서진, 김신아
검토자	김예은

목차

I. 논문 주제

Windows 환경에서 수집 가능한 아티팩트 유형별 분석 기법을 기존 연구와 비교하고, 디지털 아티팩트 수집 및 분석을 위한 통합 프레임워크를 제안한다.

II. 연구 필요성

1. 해당 프로그램 선정 이유

(1) 통화, 채팅, 파일 공유, 서버/채널 구조 등 다양한 기능이 포함된 프로그램이므로 다양한 아티팩트(로그, 캐시, 미디어 파일, 메신저 파일 등)가 생성된다.

(2) 익명성이 보장되며 실시간 소통이 가능하다는 점과 파일 공유, 서버 대체 통신 수단, 범죄 모의 등의 수단으로 사용이 가능하므로 악용될 가능성이 높다. 따라서 디지털 포렌식 관점에서 주목해야 할 프로그램이라고 판단한다.

(3) 전 세계에서 많이 사용하는 메신저로 게임, 개발 등 다양한 목적을 가진 사람들이 사용하므로, 분석에 높은 가치가 있다고 판단한다.

2. 기존 연구 요약 및 한계

기존의 Discord 를 대상으로 한 디지털 포렌식 연구들은 로컬 캐시 및 로그 데이터를 중심으로 한다. 일부 연구에서는 운영체제별 비교 분석이나 포렌식 자동화 도구 개발, 또는 메모리와 네트워크 아티팩트를 대상으로 한 분석도 존재한다.

이러한 연구들은 Discord 에서 추출 가능한 디지털 아티팩트를 식별하고, 포렌식 도구의 효과성을 검증하는 데 기여한다.

그러나, 이러한 연구들은 Discord 아티팩트를 체계적으로 분류하거나 표준화된 분석 절차로 통합하려는 시도는 부족하다는 한계가 존재한다.

3. 개선 방향

분석 결과의 재현성, 정량성, 법적 활용 가능성을 포괄적으로 고려한 디지털 아티팩트 분석 프레임워크를 제안한다.

아티팩트의 저장 위치, 포맷, 분석 방법 등의 연계성을 확보하여 실제 수사 환경에서의 적용이 가능하도록 한다.

특히, Discord 내에 존재하는 다양한 디지털 아티팩트 유형별로 가장 효과적인 분석 접근 방안을 제시한다.

III. 연구 목적

1. 디스코드 아티팩트 분석에 대한 방법론 제시

본 연구의 목적은 Discord 프로그램에서 생성되는 다양한 아티팩트를 체계적으로 분석하고 이에 대한 구체적인 분석 방법론을 제시하는데 목적이 있다.
각 아티팩트 유형에 적합한 분석 방법을 제안함으로써 수사 현장에서 활용 가능한 일관된 분석 프레임워크를 구축하고자 한다.

2. 기존 논문과의 분석 방법 비교

기존의 Discord 관련 포렌식 연구들이 로그 및 캐시 분석에 집중된 것과 달리 본 연구에서는 **다양한 위치에 분산되어있는 아티팩트를 수집** 및 **분석**하였으며, 이를 통해 기존 연구의 분석 방법과의 차이를 비교하고자 한다.

IV. 연구 방법

1. 아티팩트 비교 분석 연구

1) 메신저 아티팩트

(1) 기존 연구와 동일한 내용 : 내부의 leveldb 파일 분석을 통한 활동 로그에 대한 정보

(2) 기존 연구와의 분석 방법 차이:

(3) 새로 알게 된 내용:

① C:₩Users₩DISCORD-TEST PC₩AppData₩Roaming₩discord₩LocalStorage₩ 에서 로그인 정보를 확인할 수 있다.

②

C:₩Users₩DISCORD-TESTPC₩AppData₩Roaming₩discord₩logs₩renders_js.log 에서 확인할 수 있는 서버 및 채널 이동 정보, 메시지 및 파일 전송 정보는 논문에서 자세하게 서술되어있지 않기 때문에 **해당 로그에서 업로드 모듈, 파일 정보를 타임스탬프와 비교 분석하는 방법을 세부적으로 정립하는 방법을 제시**한다.

③ 서버 및 채널 이동, 메시지 전송 및 파일 전송, 로그인 정보에 대한 로컬 캐시와 로그 데이터를 확인할 수 있었다.

2) 파일 사용/조작 아티팩트

(1) 새로 알게 된 내용:

사용자 계정에 관련된 정보를 확인하기 위하여 MFT, USN Journal, NTFS 로그 파일을 추출한 뒤, 해당 데이터가 store.db 에 저장된다는 사실을 확인할 수 있었다.

해당 DB 의 schema 와 data 에서는 공급자 정보, GUID 정보, 타임스탬프, 이벤트 로그, 이벤트 성공 여부 등을 확인이 가능하다.

store.db 를 분석한 내용은 기존 논문에서도 서술되어있지 않기 때문에, **db 파일에 저장된 데이터와 메신저 아티팩트에서 확인한 사용자 기록을 비교하는 분석하는 방법을 제시**한다.

```
-- Schema: UPDATESPROP ---
CREATE TABLE UPDATESPROP ( PROVIDERID TEXT NOT NULL COLLATE NOCASE CHECK(PROVIDERID <> ''), UPDATEID TEXT NOT NULL COLLATE NOCASE CHECK(UPDATEID <> ''), VARIABLE TEXT NOT NULL COLLATE NOCASE CHECK(VARIABLE <> ''), VALUE TEXT, TYPE INTEGER, PRIMARY KEY(PROVIDERID, UPDATEID, VARIABLE) FOREIGN KEY(PROVIDERID, UPDATEID) REFERENCES UPDATES(PROVIDERID, UPDATEID) ON DELETE CASCADE);
--- Data: UPDATESPROP ---
('LegacyUOProvider', 'ia1339254110005679 98', 'CorrelationVector', 'TCY3wFnxukq5MAN8.0', 4)
('LegacyUOProvider', 'ia1339254110005679 98', 'DiscoveryTime', '1748067502993', 3)
('LegacyUOProvider', 'ia1339254110005679 98', 'QueueNumber', '1', 2)
('WuProvider', '3f7f91a3-b4ac-4823-8050-046a7743f4e9:200', 'DiscoveryTime', '1748074789355', 3)
```

```
('WuProvider', '3f7f91a3-b4ac-4823-8050-046a7743f4e9:200', 'QueueNumber', '2', 2)
('WuProvider', '6500af43-533e-41f8-8413-f70648777d13:201', 'DiscoveryTime',
```

[store.db 덤프 화면 중 일부]

3) 네트워크 아티팩트

(1) 새로 알게 된 내용:

네트워크 아티팩트에 대한 분석을 통해 세션 토큰, 관련 도메인, 네트워크 로그 기록, 쿠키 정보, 사용자 디바이스 IP 정보 HSTS 정책 정보를 확인할 수 있었다.

사용자 디바이스 IP 정보는 C:₩Users₩DISCORD-TEST PC₩AppData₩Roaming₩discord₩Network₩Network Persistent State 경로에서 확인할 있으며, 다음과 같은 JSON 구조를 통해 확인 가능하다.

```
{"address":"192.168.116.133","used_quic":true}
```

[Network Persistent State 에서 확인한 디바이스 IP 정보]

서버 연결 정보 또한 동일한 Network Persistent State 파일 내에서 확인 가능하며, 연결된 서버의 주소, 사용 프로토콜, 응답 시간 등의 정보를 포함한다.

```
{"net":{"http_server_properties":{"servers":[{"anonymization":[],"network_stats":{"srtt":45108}
```

[Network Persistent State 에서 확인한 서버 연결 정보]

HSTS 정책 정보는 C:₩Users₩DISCORD-TEST-PC₩AppData₩Roaming₩discord₩Network₩TransportSecurity 경로에서 확인할 수 있으며, 캐시된 HSTS 정책의 호스트, 만료 시간, 서브도메인 포함 여부 등의 정보가 포함되어 있다.

```
[{"expiry":1779602913.006298,"host":"DChOn9dTAFRZISLL+cChKGqikCvagkbgxpDyL7z4je8=","mode":"force-https","sts_include_subdomains":true)
```

[TransportSecurity 에서 확인한 HSTS 정책 정보]

이러한 사용자 디바이스 IP 정보, 서버 연결 정보, HSTS 정책 정보는 기존 연구들에서 상세히 다루어지지 않은 부분이므로, 본 분석에서는 **각 경로에서 해당 정보를 추출하고 이를 기반으로 네트워크 활동을 분석하는 방법을 새롭게 제시**한다.

2. 프레임워크 제시

1) 분석 환경: Windows 10 home
2) 분석 도구: FTK Imager, NTFS Log Tracker, Volatility 등
3) 이미지 파일 형식: vmdk, E01 파일
4) 분석 대상: 설치 정보, 로컬 저장소, 설정 및 로그 파일, 레지스트리, 파일 시스템 등
5) 분석 경로: %AppData%₩Roaming₩discord 디렉터리의 로그, 캐시 세션 정보 등, $MFT, $UsnJrnl, $LogFile, NTUSER.DAT 등 파일 시스템 메타데이터
6) 분석 내용: 사용자 행위, 메시지 내용, 파일 첨부/전송, 삭제 행위 여부, 타임라인 분석 등 (중복 데이터 제거 및 무관한 파일 필터링)
7) 분석 기준: 프로그램 설치 및 실행, 메시지 작성/삭제, 특정 기능 사용 여부, 접근 파일 및 활동 흔적 등
8) 결과 문서화 : 보고서 형태의 산출물

분석 대상	경로 및 위치	분석 도구 및 방법
Discord 설치 정보	%AppData%₩Local₩Discord%LocalAppData%₩Discord	파일 시스템 분석(FTK, Autopsy), 설치 시간 확인(Prefetch 등)
계정 정보 및 로그인 기록	%AppData%₩Roaming₩discord₩Local Storage₩leveldb	leveldb 분석, 로그 파일 내 로그인/세션 추적(leveldb viewer)
채팅 메시지 및 기록	%AppData%₩Roaming₩discord₩Local Storage₩leveldb	문자열 추출 및 타임라인 분석 (leveldb viewer)
파일 및 링크 기록	%AppData%₩Roaming₩discord₩Cache	캐시/다운로드 파일 해시 비교 (FTK, Autopsy)
역할 부여 및 설정 변경	%AppData%₩Roaming₩discord₩Local Storage₩leveldb (키워드 role, permissions 등 검색)	키워드 기반 분석, 이벤트 흐름 타임라인 정리 (LevelDB, 메시지 로그)
프로필 변경 기록	%AppData%₩Roaming₩discord₩Cache %AppData%₩Roaming₩discord₩Local	캐시 파일 비교 및 타임스탬프 추적

	Storage₩leveldb	
이벤트 생성 기록	%AppData%₩Roaming₩discord₩Local Storage₩leveldb	타임스탬프 기반 타임라인 작성 및 분석
서버 삭제 및 계정 삭제 기록	%AppData%₩Roaming₩discord 하위 전체	계정 삭제 요청 기록 및 프로세스 분석 (Volatility, evtx 등)
시스템 파일 메타데이터	$MFT, $LogFile, $UsnJrnl, NTUSER.DAT	특정 파일의 접근 시점 추적, 파일 시스템 타임라인 재구성

[표 1. 분석 방법 요약 표]

유형	경로	도구	분석 항목	기존 연구와 차별성
메신저	logs/LocalStorage 폴더	FTK Imager	서버 및 채널 이동, 메시지 로그	renderer_js.log 분석 심화
파일 조작	store.db, $MFT 폴더	LEVELDB	사용자 계정 정보	로그/DB 비교 기반 근거 제시
네트워크	Network 폴더	HxD, Wireshark	내부 IP, HSTS 정책 적용 여부	HSTS, QUIC 분석 논문 부재

[표 2. 비교 분석 내용]

V. 연구 결과

1. 비교 분석 내용 및 표
2. 프레임워크 제시

VI. 기대 효과

1. 디스코드 아티팩트 분석에 대한 정확도 제고

기존 Discord 분석 논문에서는 IndexedDB·LevelDB 내부 구조, 음성 메시지 전송·삭제 로그, 모바일 앱 전용 캐시 등 일부 아티팩트를 부분적으로만 조명하거나 누락하는 경우가 존재한다. 본 연구에서는 아티팩트에 대한 분석

범위를 확장하여 기존 논문에서는 자세히 다루지 않은 아티팩트를 비교하며 디스코드 어플리케이션 분석에 대한 정확도를 높일 수 있다.

2. 워크플로우 표준화

본 연구에서는 Discord 아티팩트 수집, 분석에 이르는 과정을 일관된 절차로 정의한 통합 프레임워크를 제안한다. 이를 통해 후속 연구자나 수사관이 동일한 환경과 방법으로 연구를 재현할 수 있으며, 분석 결과 간 비교와 검증이 용이해져 연구의 신뢰성이 향상될 수 있다.

VII. 연구 한계

1. 각 아티팩트를 분석할 때 생기는 한계에 대해서 서술

(1) 메신저 아티팩트

Discord 는 클라이언트-서버 구조를 기반으로 하여 메시지를 클라우드에 저장하는 방식을 취하고 있어, 로컬 저장소 남아 있는 정보만으로는 모든 메시지의 전송, 수신, 삭제 여부를 완전하게 분석하는 데에 한계가 있다. 사용자가 메시지를 삭제하거나 계정을 비활성화할 경우 관련 데이터가 완전히 제거될 수 있으며, 데이터 복구가 어렵다. 일부 메시지 데이터는 암호화되어 있거나 구조가 비정형적으로 저장되어, 정확한 시간 정보나 맥락을 파악하는 데 한계가 따른다.

(2) 파일 사용/조작 아티팩트

store.db 는 GUID, 이벤트 로그, 타임스탬프 등 다양한 정보를 포함하고 있어 유용한 분석 자료가 되지만, 운영체제나 파일 시스템의 종류에 따라 동일한 아티팩트라도 저장 위치나 기록 방식이 달라질 수 있어, 분석 방법의 표준화나 일반화가 어렵다. 암호화 기술이 광범위하게 적용되면 내부 데이터에 직접 접근하기 어려워지는 경우가 많아, 분석의 한계가 따른다.

(3) 네트워크 아티팩트

사용자의 접속 기록이나 쿠키, 세션 로그 등을 통해 활동 경로와 환경을 파악할 수 있으나, 대부분의 통신이 암호화 프로토콜을 통해 보호되기 때문에 통신 내용을 직접 분석하는 것은 불가능하다. 일부 로그는

불완전하거나 누락되어 있을 수 있어 IP 나 디바이스 정보만으로는 사용자를 명확히 식별하기 어렵다는 점에서 분석의 신뢰성과 정확성에 한계가 따른다.

2. 기존 연구와의 차이점이나 동일함에 대한 한계 서술

기존의 Discord 분석 논문들에서 다루지 않았던 store.db 등 일부 새로운 아티팩트를 포함하여 분석 범위를 확장하고자 한다. 기존 연구와 본 연구가 유사한 결과를 도출하는 경우, 해당 유사성이 도구의 동일성에 의한 것인지, 환경 설정이 유사했기 때문인지 명확하게 구분하기 어려워 분석 결과 해석에 혼란을 줄 수 있다. 이러한 점에서 기존 연구와의 차별성을 강조하는 데 한계가 존재한다.

VIII. 참고 문헌

[1] 신수민, 박은후, 김소람, 김종성,「디지털 포렌식 관점에서의 Slack 및 Discord 메신저 아티팩트 분석」, 디지털콘텐츠학회논문지 제 21 권 제 4 호, 2020.4, 799-809.

[2] Michał Motyliński, Áine MacDermott, Farkhund Iqbal, Mohammed Hussain, Saiqa Aleem,「Digital Forensic Acquisition and Analysis of Discord Applications」, 2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), 2020.11.

[3] Farkhund Iqbal, Michał Motyliński, Áine MacDermott,「Discord Server Forensics: Analysis and Extraction of Digital Evidence」, 2021 11th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2021.5.

[4] Kyle Moffitt, Umit Karabiyik, Shinelle Hutchinson, Yung Han Yoon,「 Discord Forensics: The Logs Keep Growing」, 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), 2021.1.

[5] Muhammad Koprawi, Fadhli Dzil Ikram「Forensic analysis on discord application using the National Institute of Standards and Technology (NIST) Method」, Jurnal Mandiri IT Vol. 12 No. 1 (2023): July: Computer Science and Field, 2023.8.

[6] Khushi Gupta, Phani Lanka, Cihan Varol,「A holistic digital forensic analysis of Discord – Storage, memory, and network perspectives」, Journal of Forensic Sciences: Volume 69, Issue 4, 2024.6, 1320-1333.