라이브 포렌식을 통한 디지털 증거 수집 구현

이 원 희[1], 이 지 훈[1], 안 채 혁[1], 우 수 민[1], 신 상 욱[2]
[1]부경대학교 컴퓨터·인공지능공학부 학부생
[2]부경대학교 컴퓨터·인공지능공학부 교수

NexusX4444@gmail.com, huny10000@naver.com, theresia0168@gmail.com, superb_m98@pukyong.ac.kr, shinsu@pknu.ac.kr

Implementation of Digital Evidence Collection through Live Forensics

Won-hui Lee[1], Ji-hoon Lee[1], Chae-hyeok Ahn[1], Su-min Woo[1], Sang Uk Shin[2]
[1]Div. of Computer Engineering and artificial intelligence, Pukyong National University
[2]Div. of Computer Engineering and artificial intelligence, Pukyong National University

요 약

본 연구는 사용자가 USB에 내장된 스크립트를 실행하여 실시간으로 활성 및 비활성 데이터를 수집하는 라이브 포렌식 도구의 개발에 관한 것이다. 이 도구는 컴퓨터에 USB를 삽입하고 특정 스크립트를 실행하여 중요한 디지털 증거물을 추출하고 분석하는 기능을 제공한다. 도구는 Linux와 Windows 운영 체제용 32비트 및 64비트 버전으로 제작되었으며, 대량의 데이터 처리 시간과 저장 공간 문제를 해결하여 필요한 특정 데이터만 신속하게 추출할 수 있는 효율적인 방법을 제공한다. 이 도구는 활성 데이터와 비활성 데이터를 수집하며, 활성 데이터에는 레지스터, 네트워크 정보, 프로세스 정보, 사용자 정보 등이 포함되며, 비활성 데이터에는 메타데이터, 시스템 설정 정보, 로그 파일 등이 포함된다. 이 연구에서는 라이브 포렌식 도구의 사용 방법과 수집된 결과, 데이터 분석 방법, 그로 인한 보안 이점에 대해 다루고 있다.

1. 서 론

본 연구는 컴퓨터 시스템에서의 라이브 포렌식을 중요한 역할로 간주하며, 기존 라이브 포렌식 도구의 한계를 지적한다. 기존 도구는 운영체제 버전에 따라 동작하지 않거나 큰 저장 공간과 많은 시간이 필요한 문제가 있다. 이에 대한 대안으로 USB에 내장된 스크립트를 실행하여 실시간으로 필요한 데이터만을 추출하고 분석하는 통합 라이브 포렌식 도구를 개발했다. 이 도구는 Linux/Windows 운영체제를 지원하며, 사용자가 USB를 삽입하고 스크립트를 실행하여 빠르게 데이터를 추출하고 분석할 수 있도록 도와준다. 본 논문은 이 도구의 사용 방법, 효과, 그리고 라이브 포렌식 분야에서의 활용 가능성에 대해 논의하며, 새로운 관점과 해결책을 제시하여 실무자들에게 도움을 주고자 한다.

2. Unified forensic Toolkit(UFT)

2.1. 프로그램 개요

본 논문에서는 Windows와 Linux 각각의 운영체제에서 라이브 포렌식을 수행하기 위해 개발한 스크립트에 대하여 서술한다. 해당 스크립트들은 각 운영 체제에 특화되어 있으며, 활성 데이터와 비활성 데이터의 수집에 사용된다. 본 포렌식 프로그램은 USB에 내장된 형태로 이루어져 있으며, 두 개의 파티션으로 분할되어 있다. 각 파티션은 프로그램을 실행하는 파티션과 포렌식 과정을 통해 수집한 데이터 및 결과물을 저장하기 위한 파티션으로 분할되어 있다. 마지막으로 지금까지 수집한 결과물을 간략하게 설명하는 리포팅 툴을 추가적으로 제작했다.

2.1.1. 공통 사항 개요

2.1.1.1. 수집 대상 데이터 분류

포렌식을 위한 수집 대상 데이터는 크게 활성 데이터와 비활성 데이터를 기준으로 분류한다. 활성 데이터는 RFC 3227에 따른[1] 휘발성 기준으로 레지스터, 네트워크 정보, 프로세스 정보, 로그온 사용자 정보, 시스템 정보, 자동실행 프로그램, 클립보드/작업 스케쥴러 정보 순서로 수집한다. 비활성 데이터는 파일 시스템 메타 데이터, 시스템/사용자/어플리케이션 설정 정보, 프리패치, 로그 파일, 휴지통 정보, 웹 히스토리 정보, 임시 파일, 시스템 복원 지점, 외부 저장 매체, 바로가기 파일의 순서로 우선순위를 정하여 수집하도록 하였다[2].

2.1.1.2. 결과물 무결성 확인

스크립트를 통해 수집한 데이터에 대해 무결성을 증명하기 위하여 수집 당시 타임 스탬프, 해시값을 기록한다. 타임 스탬프는 시스템의 변화가 발생할 수 있을 경우 기록하도록 하였다. 또한, 결과물에 대한 해시값을 기록하여 수집 이후 무결성 증명한다.

2.1.2.3. 결과물 보고서 생성

지금까지 얻은 결과물을 바탕으로 리포트를 생성하여 대상 시스템에 대한 상태정보를 간략하게 확인할 수 있도록 지원한다.

2.2. 개발환경 개요

Windows 개발환경은 주로 Windows10/11 기본 쉘 스크립트를 사용하여 데이터를 신속하게 수집하기 위해 OS 내장 명령어 및 Sysinternals, Nirsoft 도구 및 물리 메모리 덤프를 얻기 위해 참고자료를[3] 활용하였다. 또한, 가능한 기본 명령어를 활용하여 윈도우 운영체제 간의 호환성을 고려하였다. Linux 개발환경에서는 VMware Workstation을 활용하여 Ubuntu Linux 64bit 22.04.2 LTS 버전을 이용했다. Bash shell 스크립트를 통해 프로그램을 작성하였으며, Avml을 이용한 메모리 덤프 작업을 위해 Rust를 usb에 설치하였다. 뿐만 아니라, 결과물의 해시값 추출을 위하여 C 기반의 해시 추출 프로그램을 사용하였다.

2.2.1.1. Windows/Linux 동작 원리

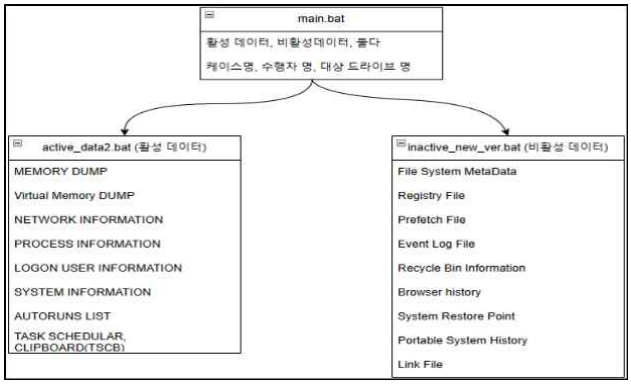

그림 1. 전반적인 Windows 스크립트 구성

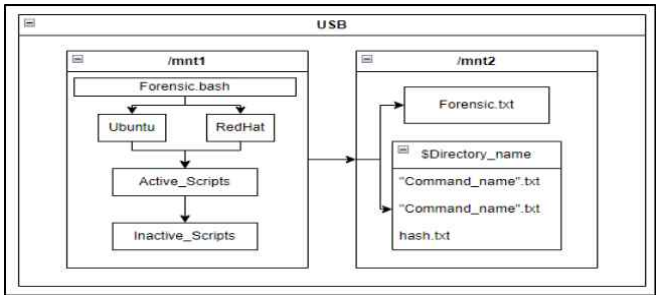

그림 2. 리눅스 스크립트 구조

메인 스크립트를 처음 실행했을 시 조사관은 사건 케이스명과 조사관의 성함을 작성하고 앞으로 라이브 포렌식 데이터가 저장될 디렉토리가 어디인지 명시해야 합니다. 지금까지 입력한 각 값들이 활성/비활성 스크립트 실행할 때 인자로 전달되며 해당 인자는 추후에 덤프 생성시 사용될 디렉토리 절대경로 및 이후에 보고서를 만들 때 사용합니다. 사용자가 복수의 옵션을 선택하여 각 파트별로 데이터를 수집하고 난 후 덤프한 데이터를 바탕으로 보고서를 작성하는데. 보고서는 지금까지 어떤 데이터를 수집했고 해당 데이터의 타임 스탬프 및 해시값이 어떤지 간략하게 나타내는 형식으로 생성합니다.

3. 결 론

본 논문에서는 라이브 포렌식을 위한 두 가지 스크립트를 소개한다. 이들 스크립트는 Windows와 Linux 운영 체제에 특화되어 있으며, 활성 데이터와 비활성 데이터를 효과적으로 수집할 수 있다. 또한, 이들 스크립트는 사용자가 시스템의 상태를 신속하게 파악하고, 필요한 정보를 즉시 수집할 수 있도록 도와준다. 이 스크립트들은 신뢰할 수 있는 도구와 운영 체제 내장 명령어를 사용해 작성했으며, CLI(Command Line Interface)를 사용하여 빠르고 효율적으로 실행될 수 있다. 또한, 이들은 x86/x64 아키텍처를 지원하며, 수집된 데이터의 타임스탬프와 해시 값을 저장하여 데이터의 무결성을 보장한다. 또한 수집한 결과물을 토대로 리포트를 자동 생성하여 실무진이 대상 시스템에 대한 상태를 간단히 확인할 수 있도록 돕는다. 하지만 특정 시스템 설정이나 환경에서는 예상치 못한 문제가 발생할 수 있다. 또한, 이 스크립트들은 현재 가장 널리 사용되는 운영 체제인 Windows와 Linux에만 특화되어 있으며, 특히 일부 에서는 특정 명령이 지원되지 않을 수 있으므로 더 다양한 상황을 고려하여 세밀하게 스크립트를 작성할 필요가 있다. 마지막으로, 라이브 포렌식을 진행하는 과정에서는 사용자가 USB를 장착하고 스크립트를 수행하는 행위나 사용자가 침해사고를 늦게 인지하거나 자체적으로 조사를 진행하면서 원본 데이터의 무결성이 어느 정도 손상될 수 있다는 점을 고려해야 한다. 결론적으로, 본 논문에서 제시한 스크립트들은 라이브 포렌식의 중요성을 잘 나타내고 있으며, 이 분야의 연구와 개발에 중요한 기여를 하였다고 생각한다.

참고문헌

[1] "Guidelines for Evidence Collection and Archiving, RFC 3227"

[2] NIST, “Guide to Integrating Forensic Techniques into Incident Response”, Special Publication 800-86, 5-1

[3] https://belkasoft.com/ram-capturer