情報保護學會論文誌
第16卷 第1號, 2006. 2

윈도우 시스템에서 디지털 포렌식 관점의 메모리 정보 수집 및 분석 방법에 관한 고찰*

이 석 희,[1†] 김 현 상,[1] 이 상 진,[1‡] 임 종 인[1]
[1]고려대학교 정보보호대학원

A Study of Memory Information Collection and Analysis in a view of Digital Forensic in Window System*

Seok-Hee Lee,[1†] Hyun-Sang Kim,[1] SangJin Lee,[1‡] JongIn Lim[1]
[1]Center for Information Security Technologies, Korea University

요 약

본 논문에서는 RFC3227 문서[1]에 따른 일반적인 디지털 증거 수집 절차를 살펴보고 메모리 정보 수집에 대한 절차를 정립하였다. 또한 디지털 증거 수집 절차의 개선 사항으로 메모리 덤프 절차를 포함시키고 시스템 전체 메모리를 획득하는 방법을 제시하였다. 실 사용자들의 메모리를 덤프하고 가상 메모리 시스템의 페이지 파일을 수집하여 얼마나 많은 정보가 검출되는 지를 확인해 보았는데, 이 과정에서 핵심 보안정보인 사용자 ID와 패스워드가 페이지 파일의 절반 정도에서 검출되었으며 데이터복구를 통해 중요 정보를 획득할 수 있음을 확인하였다. 또한 각각에 대한 분석기법과 메모리 정보를 중심으로 하는 메모리 정보 획득 절차를 제시하였다.

ABSTRACT

In this paper, we examine general digital evidence collection process which is according to RFC3227 document[1], and establish specific steps for memory information collection. Besides, we include memory dump process to existing digital evidence collection process, and examine privacy information through dumping real user's memory and collecting pagefile which is part of virtual memory system. Especially, we discovered sensitive data which is like password and userID that exist in the half of pagefiles. Moreover, we suggest each analysis technique and computer forensic process for memory information and virtual memory.

Keywords : *Computer Forensics, Virtual Memory, Pagefile*

Ⅰ. 서 론

최근 컴퓨터에 저장되어 있는 데이터가 법정에서 다루어지는 경우가 많은데 이와 관련된 분야를 컴퓨터 포렌식이라고 하고, 정보처리 기기를 통하여 이루어지는 각종 행위에 대한 사실관계를 확증하거나 증명하기 위해 행하는 각종 절차와 방법으로 정의할 수 있다[15]. 컴퓨터 포렌식은 크게 시스템 포렌식과 네트워크 포렌식으로 분류할 수 있다. 시스템 포렌식이란 대상 컴퓨터 시스템에 남아있는 흔적을 가

접수일 : 2005년 10월 19일 ; 채택일 : 2006년 1월 12일
* 본 연구는 정보통신부 대학 IT 연구센터 육성·지원 사업의 연구 결과로 수행되었습니다.
† 주저자 : gosky7@korea.ac.kr
‡ 교신저자 : sangjin@korea.ac.kr

공, 수집, 분석, 보관하는 과정이며, 네트워크 포렌식이란 침해사고 확인을 시발점으로 침해와 관련 있는 네트워크 이벤트를 수집, 분석, 저장하는 일련의 과정이다[16][14]. 포렌식 절차는 포렌식 준비, 증거물 획득, 증거물 보관 및 이송, 증거 분석 단계, 보고서 작성 단계로 구성되어 있다. 이 중 증거물 획득 단계는 사고 발생 현장에서 디지털 증거를 수집하고, 증거의 무결성을 확보하는 단계이다[16]. 일반 수사에서 현장을 확인하고 관련 증거를 수집하는 초등 수사가 정확하고 효율적으로 수행되기 위해서는 이 단계가 가장 중요한 부분이다[17]. 현재 저장 매체의 디지털 증거 수집 절차로 잘 알려진 표준으로 RFC 3227(Guidelines for Evidence Collection and Archiving)[1]이 있다. 이 문서에서 제시된 수집 절차는 다음과 같다.

- 사고와 관련되어 있는 시스템과 수집해야 할 증거들을 목록화 한다.
- 증거로써 인정될 수 있는 것들을 결정한다. 증거물로써 인정될지 아닐지 확실하지 않은 사항일 경우에는 수집 목록에 되도록 많은 사항을 포함시킨다.
- 각 시스템 별로 휘발성이 강한 정보들부터 수집 순서를 정한다.
- 증거가 변경될 수 있는 외부 요소들을 제거한다.
- 휘발성이 강한 순서에 따라서 증거 수집 도구를 사용해서 증거를 수집한다.
- 소요 시간을 기록한다.
- 수집한 것 이외에 어떠한 것들이 증거가 될 수 있는지 다시 검사한다.
- 각 단계별로 진행과정을 기록한다.
- 디지털 증거 수집 작업을 한 사람들의 모든 행동을 기록한다. 누가 거기에 있었고, 무엇을 하였으며, 무엇을 획득하였고, 어떤 작업을 했는지 등을 기록한다.
- 수집된 증거들에 대해서 체크섬 값과 전자서명 값을 생성한다.

RFC 3227은 디지털 증거물 수집에 대한 기준을 제시하였다는데 의미가 있다. 그러나 각 단계별 세부 절차가 없기 때문에 디지털 증거 수집 현장에 바로 적용할 수 없다. 본 논문에서는 휘발성 정보에서 메모리 수집 절차에 대한 일반적인 내용을 확인하고, 메모리 정보와 페이지 파일 수집에 대한 구체적인 디지털 증거 수집 절차와 분석 방법을 제시한다.

1.1 라이브 시스템의 디지털 증거 수집

라이브 시스템이란 컴퓨터 시스템이 부팅된 후 운영체제에 의해서 동작되고 있는 온라인 상태의 시스템을 말한다. 라이브 시스템은 대상 시스템의 운영 상태 및 데이타를 각종 저장 매체에 기록하고 있다. 이 정보들 중 방치하거나 시스템이 종료되면 사라지는 정보를 휘발성 정보라 한다[16]. 이 휘발성 정보는 주로 메모리 또는 하드디스크의 임시파일에 저장되어 있다. RFC3227에서는 휘발성이 강한 정보의 순서를 다음과 같이 제시하고 있다[1].

① 레지스터, 캐쉬
② ARP 캐쉬, 프로세스 테이블, 커널, 메모리, 라우팅 테이블
③ 임시 파일 시스템
④ 디스크
⑤ 의심스러운 원격 로그인과 시스템에 대한 모니터링
⑥ 물리적인 설치 상태와 네트워크 구성
⑦ 저장 매체

특히 ①, ②번 정보는 시스템이 종료될 경우 복구할 방법이 없다. 따라서 시스템 활성 상태에서 프로세스와 시스템 상태를 검사할 수 있는 프로그램을 사용하여 디지털 증거를 수집해야 한다.

디지털 증거 수집 시 사용하는 프로그램은 무결성이 보장되어야 한다. 침해가 일어난 컴퓨터 시스템의 명령어는 공격자에 의해서 변조되었을 가능성이 있기 때문에 증거 수집에는 사용할 수가 없다. 따라서 디지털 증거 수집 툴들은 따로 준비해야 하며, 이러한 프로그램들은 Forensic CD[2]에서 실행하는 것이 권장된다.

Forensic CD란 대상 시스템이 동작 중에 있는 경우에는 CD 안에 초등 수사에 필요한 프로그램을 탑재하고 있어 스크립트를 이용해 그 즉시 시스템의 휘발성 정보를 수집할 수 있고, 대상 시스템이 종료되어 있는 상태라면 하드 디스크에 인스톨할 필요 없이 CD-ROM 한 장으로 부팅해서 Forensic 작업을 할 수 있는 Live Linux CD를 말한다.

Forensic CD의 내부 실행파일은 실행에 필요한 라이브러리나 DLL 모듈을 모두 포함하는 정적 링크 방식으로 컴파일되어 있어야 하며, bat 또는 Shell script를 사용하여 단계적으로 실시되어야 한다.

표 1. 휘발성 정보 수집 스크립트 예

수행 스크립트	설 명
Time /t Date /t	시스템 날짜와 시간(시작 시간)
Psloggedon	현재 로그온 되어있는 사용자 목록
Dir /t:a /o:d /a /s c:\ Dir /t:a /o:d /a /s d:\	파일 시스템의 Time/date stamp
Netstat - na	현재 열려 있는 소켓 목록
Fport	열려진 소켓에 대기하고 있는 어플리케이션
Pslist	현재 동작하는 프로세스 목록
Nbtstat - c	현재 또는 최근 시스템에 연결했던 시스템들의 목록
Time /t Date /t	시스템 날짜와 시간(종료 시간)
Doskey /history	작업 히스토리 기록

II. 메모리 정보 수집 절차와 분석 기법

2.1 메모리 정보 수집의 필요성

초등 수사에서는 일반적으로 스크립트를 작성해서 휘발성 정보를 수집한다. 일반적으로 시스템 구성 상태 위주의 수집이 이루어지고 있다. 표 1은 윈도우 2000 및 XP 시스템에서 사용가능한 휘발성 정보를 수집하는 스크립트의 예제이다[12].

표 1에서 보는 바와 같이 휘발성 정보를 수집하는 작업 내용 중에 메모리 정보를 획득하는 절차는 포함되어 있지 않다. 하지만 메모리에는 시스템 명령어로 수집할 수 없는 많은 정보들이 포함되어 있고, 수사의 실마리를 풀 수 있는 단서를 가지고 있을 가능성이 매우 높다. 따라서 메모리 정보를 반드시 획득해야 할 필요성이 있다.

메모리 정보는 크게 시스템 메모리와 프로세스 메모리로 구분할 수 있다. 본 절에서는 프로세스 메모리를 획득하는 방법을 제시하고 프로세스 메모리에 어떠한 정보가 존재하는지를 조사해 보았다. 프로세스 메모리는 그림 1과 같이 Microsoft OEM Support Tools 패키지에 있는 userdump.exe를 사용할 수 있다.[3]

메모리 덤프 파일에는 해당 프로세스가 사용했던 정보들이 있기 때문에, 사용자가 입력한 데이터 혹은 프로세스가 생성한 데이터들이 존재한다. 메모리에 정보가 어떤 식으로 존재하는지 알아보기 위하여 두 가지 실험을 하였다. 첫 번째는 보안접속을 통해 인터넷 A 포탈 사이트에 로그인을 한 후 메모리를

```
1432 msnmsgr.exe
3704 NateOnMain.exe
3468 MonSvcNT.exe
1100 V3IMPro.exe
1364 hwp.exe
2144 WINWORD.EXE
3768 IEXPLORE.EXE
3800 cmd.exe
2872 userdump.exe

D:\output>userdump.exe 3768 empas.txt
User Mode Process Dumper (Version 3.0)
Copyright (c) 1999 Microsoft Corp. All rights reserved.

Dumping process 3768 (IEXPLORE.EXE) to
D:\output\empas.txt...
The process was dumped successfully.
```

그림 1. 메모리 덤프 과정

덤프하였고, 두 번째는 인터넷 뱅킹 프로그램을 사용한 후 해당 프로세스 메모리를 덤프하였다[3].

메모리 덤프 파일에는 바이너리 값과 문자열이 섞여 있어 수사관이 분석하기 힘들기 때문에 가독성이 편한 형태로 변환하였다. 리눅스에서는 오브젝트 파일이나 실행 파일에서 ASCII 코드나 유니코드 문자열을 검색해서 출력해 주는 'strings'명령을 사용할 수 있고, 윈도우에는 이와 같은 명령어가 존재하지 않지만 Sysinternals사에서 이러한 기능을 하는 유틸리티를 제공하고 있다[4].

· 인터넷 익스플로러

인터넷 익스플로러를 이용해서 특정 웹사이트에 보안 접속을 통한 로그인을 하고 그 메모리를 덤프하였다. 그리고 덤프 파일에 'passwd'라는 문자열로 검색을 수행하였다. 그 결과, 파일에는 사용자의 ID와 패스워드가 평문으로 존재하는 것을 확인할 수 있었다.

```
pu=http%3A%2F%2Flogin.empas.com%2Flogin%2Flog_empas.html%3Fes%3Dhttp%253A%252F%252Fwww.empas.com&bSecure=1&userid=IDSAMPLE&passwd=PASSWDSAMPLE&x=20&y=19
```

그림 2. 인터넷 익스플로러 메모리 덤프 파일의 검색결과 (로그인 ID/패스워드)

· 인터넷 뱅킹 프로세스

인터넷 뱅킹을 사용하기 위해 특정 은행의 웹사이트에 공인 인증서를 통한 로그인을 하고 클라이언트 프로그램과 인터넷 익스플로러 프로세스 메모리를 덤프하였다. 덤프 파일에는 인증서 패스워드, 계좌번호, 해당 계좌의 비밀번호가 평문 형태 그대로 있는 것을 확인하였다.

()0021IB0001107798,ou=CHB,ou=personal,o=yessign,c=krcn=yessi gnCA,ou=LicensedCA,o=yessign,c=krCER.PASSWDSAMPLE

그림 3. 인터넷 뱅킹 클라이언트 프로그램 메모리 덤프 파일의 검색 결과(인증서 패스워드)

호_SEL=XXXX-XX-XXXXXXX&출금계좌번호 =XXXX-XX-XXXXXXX&비밀번호=PASSWDSAMPLE&입금계좌번호 =XXXX-XX-XXXXXXX&se_select=XXXX-XX-XXXXXXX&입금은행코드=02&이체금액=40,000&의뢰인성명=

그림 4. 인터넷 익스플로러(인터넷 뱅킹) 메모리 덤프 파일의 검색 결과

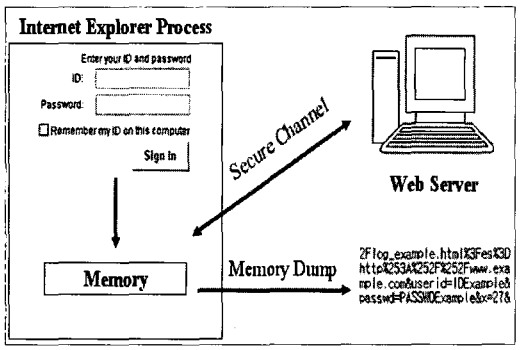


그림 5. 메모리에서의 웹 로그인 정보 추출

그림 5와 같이 웹서버에 보안접속 과정을 거쳐 로그인을 하더라도 ID와 패스워드는 반드시 메모리에 적재되었다가 암호화되어서 통신을 하기 때문에 메모리에 암호화되기 전의 평문 상태가 그대로 남아 있게 된다.

사용자가 일반적으로 웹 페이지에서 특정 정보를 입력하고 그 정보를 프로그램이 사용하기 위해서는 현재 컴퓨터 특성상 항상 데이터를 메모리에 적재하게 되어 있다. 웹브라우저의 클라이언트 프로그램이나 익스플로러의 프로세스가 서버와 암호통신을 하기 위해서는 일단 평문 데이터를 메모리에 적재하고 그것을 암호화해야 하기 때문에 컴퓨터 시스템의 특정 메모리 공간에는 사용자가 입력한 정보가 적재된다. 따라서 메모리 정보를 수집하는 것은 시스템을 종료하면 획득할 수 없는 중요한 정보를 획득할 가능성이 있다.

2.2 시스템 전체 메모리 수집

앞에서 살펴본 프로세스 메모리 덤프 방법을 사용해서는 시스템 메모리를 획득할 수가 없다. 하지만 윈도우 시스템의 최대 절전 모드를 사용한다면 시스템의 전체 메모리를 손쉽게 획득할 수 있다.

최대 절전 모드는 주로 배터리를 사용하는 랩탑 컴퓨터에서 부족한 전원을 관리하기 위해서 사용하는 기능으로써 열려 있는 모든 파일 및 문서와 함께 바탕 화면의 이미지를 저장한 다음 컴퓨터의 전원을 끈다. 즉 현재 메모리 내용을 하드 디스크에 저장하고 종료한다. 다시 전원을 켜면 전원을 끄기 전과 똑같은 작업 환경으로 부팅이 된다. 단 하드웨어에서 최대 절전 기능을 지원해야 한다는 단점이 있지만, 요즘의 대부분의 PC는 최대 절전 기능을 지원하고 있다. 이러한 최대 절전 모드는 시스템의 비활성 기간을 설정해 놓고 자동으로 사용할 수도 있고 사용자가 직접 최대 절전 모드로 전환할 수도 있다.

컴퓨터의 전원이 차단될 때 메모리의 모든 내용을 하드 디스크에 저장하는데 그 위치는 C:\hiberfil.sys라는 파일에 저장된다. 시스템이 설치될 때 운영체제에서 기본적으로 RAM 용량과 같은 크기의 공간을 확보한다. 최대 절전 기능으로 종료한 후, hiberfil.sys 파일을 획득한다면 현재 메모리 전체 내용을 획득할 수 있다.

Hiberfil.sys 파일은 해당 하드 디스크를 분리하여 다른 시스템에 슬레이브로 디스크를 장착하여 획득할 수 있고, Forensic CD를 이용하여 획득할 수도 있다. 하지만 디스크 이미징 작업이 예정되어 있다면 이미징 파일 안에서 획득할 수가 있어 또 다른 추가작업을 하지 않아도 된다.

기존의 알려진 일반적인 포렌식 절차에서는 라이브 데이터를 수집하고 난 후 시스템의 전원을 바로 차단하여 비정상 종료[11] 후 디스크 이미징 작업을 하지만 이러한 절차는 시스템 전체 메모리를 획득할 수 없고, 시스템에 치명적인 손상을 가할 위험이 있다. 최대 절전 모드는 종료될 때 메모리의 데이터를 하드 디스크로 옮기고 전원을 바로 차단하여 종료하지만 시스템에 하드웨어적인 손상을 가하지 않기 때문에 시스템 메모리 정보도 확보하고 하드 디스크의 내용도 변경하지 않는다. 따라서 디스크 이미징 작업을 할 때에는, 기존에 알려진 전원을 차단하여 비정상 종료하는 방법 대신 최대 절전 모드를 사용하는 것으로 바뀌어야 한다. 최대 절전 기능을 사용하는 방법은 표 2와 같다.

III. 가상 메모리 시스템과 페이지 파일

현재 사용되고 있는 대부분의 컴퓨터 시스템은 가상 메모리 기법을 이용한다. 가상 메모리(Virtual

표 2. 최대 절전 모드 사용법

제약사항	1. 전원 사용자 그룹의 구성원 또는 관리자로 로그온 해야 한다. 2. 컴퓨터가 네트워크에 연결되어 있으면 네트워크 정책 설정으로 인해 이 절차를 완료하지 못할 수도 있다.
순서	1. 제어판에서 전원 옵션을 연다. 2. 최대 절전 모드 탭을 클릭한 다음 최대 절전 모드 지원 확인란을 선택한다. (최대 절전 모드 탭이 표시되지 않으면 하드웨어가 이 기능을 지원하지 않는 것이다.) 3. 확인을 클릭하여 전원 옵션 대화 상자를 닫는다. 4. 시스템 종료를 클릭하고, 드롭다운 목록에서 최대 절전 모드를 선택한다.

Memory)는 운영체제에 의해 구현되는 개념으로서, 프로그래머에게 큰 용량의 메모리나 데이터 저장 공간을 사용할 수 있도록 허용하는 것을 말한다[5]. 즉 보조기억 장치를 마치 주기억 장치인 것처럼 이용하는 방법으로, 실제 존재하지 않는 메모리를 있는 것처럼 사용하는 방법이다.

프로세스가 수행되려면 주 기억 장소 내에 있어야 하지만 시분할 시스템에서 일시적으로 주 기억 장소에서 보조 기억 장소로 이주될 수 있고 수행될 필요성이 있으면 다시 주 기억 장소로 되돌아온다. 어떤 프로세스가 중앙처리장치의 사용 기간이 만료되어 더 이상 사용할 수 없을 경우 이 프로세스는 더 이상 주 기억 장소에 있을 필요가 없다. 이런 프로세스를 잠시 메인 메모리에서 보조 기억 장치의 pagefile로 이주시킨다. 이때 주기억 장치와 보조 기억 장치 사이의 데이터 교환을 스왑핑(swapping)이라고 한다. 그리고 보조 기억 장치, 즉 하드 디스크에서 주기억 장치로 들어오는 것을 swap-in, 주기억 장치에서 하드 디스크로 물러나는 것을 swap-out이라고 한다[6].

그림 6의 스왑과정은 다음과 같다.

① 프로그램 실행
② 운영체제가 메모리에서 중요하지 않다고 판단되는 프로세스 선택
③ 하드디스크의 스왑 영역으로 이동
④ 메모리에 빈 공간 확보
⑤ 프로그램 로딩

표 3. 페이지 파일의 기본 크기

시스템 메모리 크기	페이지파일의 최소크기	페이지파일의 최대크기
1GB 이하	1.5 * RAM	3 * RAM
1GB 초과	1 * RAM	3 * RAM

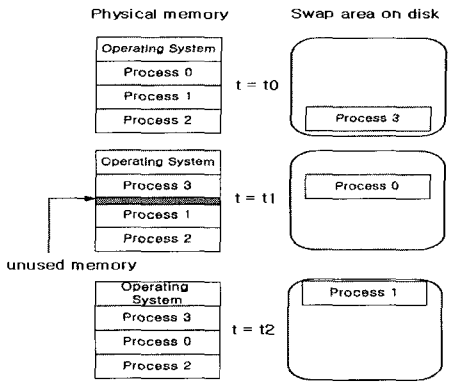


그림 6. 스왑 과정

3.1 페이지 파일 개요

윈도우 NT/2000 시스템에서 사용하는 스왑 영역은 통상 실제 메모리 용량의 1.5배 크기로 C:\pagefile.sys 파일로 만들어지고 윈도우 9x에서는 C:\windows\win386.swp 파일로 만들어진다.

그림 7은 윈도우 시스템에서 페이지 파일에 접근하는 프로세스들을 모니터링 한 것이다[7]. 윈도우 시스템에서 메인 메모리의 정보를 하드 디스크의 페이지 파일에 기록하는 모습을 볼 수 있다. 즉 메인 메모리의 일정 영역에 존재하던 특정 프로세스의 데이터가 더 이상 필요하지 않거나 작업 스케쥴링 알고리즘에 의해서 잠시 페이지 파일로 이동되는 것을 알 수 있다. 페이지 파일로 이동되는 페이지 정보는 수사 진행에 핵심적인 내용과 용의자의 개인 정보 및 사건 정황에 대한 정보를 담고 있을 가능성이 있기 때문에 페이지 파일의 내용도 조사할 필요성이 있다.

페이지 파일에 기록되어 있는 정보들은 윈도우에서 기본적으로 시스템이 종료될 때 삭제하지 않게 되어 있다. 다음의 레지스트리 정보를 통해 설정 값을 확인할 수 있다.

-HKLM\System\CurrentControlSet\Control\SessionManager\MemoryManagement\Cl

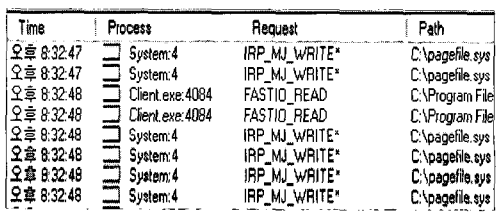


그림 7. Pagefile.sys 파일 접근 모니터링

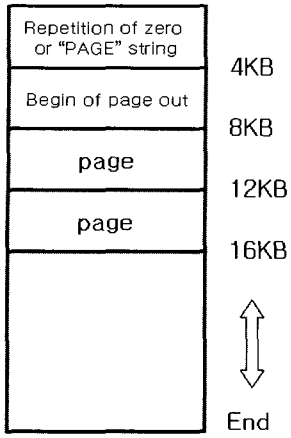


그림 8. Pagefile 구조

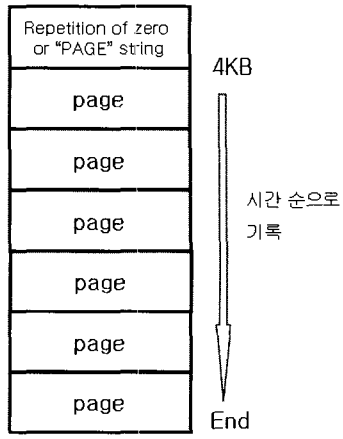


그림 9. 페이지 파일 기록 순서

earPageFileAtShutdown 의 값이 0일 경우, 종료될 때 페이지 파일의 내용을 삭제하지 않는다[8].

다음 그림 8은 페이지 파일의 대략적인 구조이다.

운영체제를 하드 디스크에 새로 설치할 경우 시스템에서 페이지 파일의 영역을 하드 디스크상 임의의 공간에 할당한다. 이렇게 할당된 페이지 파일에서 하드 디스크의 이전 사용 흔적을 찾을 수 있다. 왜냐하면 하드 디스크의 데이터를 삭제하고 포맷을 하더라도 실제 데이터는 지우지 않기 때문에 시스템을 설치하기 이전의 데이터가 남아 있기 때문이다.

페이지 파일의 첫 4096byte는 모두 "0"으로 채워지거나 "PAGE"라는 문자열을 반복해서 적는다. 첫 4096 바이트는 "0"으로 채워지는 경우가 대부분이다. 단 시스템 종료 시 페이지 파일을 삭제하도록 레지스트리 값을 설정하면, 시스템 종료 과정 중에 페이지 파일의 첫 번째 4096byte는 반드시 "PAGE"라는 문자열로 값을 설정한다. 실제 가상 메모리 정보들은 두 번째 4096byte부터 채워진다. 페이지 단위 4096byte로 page-out되는 내용들이 기록되며 페이지 파일의 끝까지 확장되면서 기록된다.

3.2 페이지 파일 분석

3.2.1 데이터 분석 기법

Disk 전체의 이미지 작업이 예정되어 있다면, 디스크 이미지 안에 해당 파일이 존재하기 때문에 페이지 파일을 따로 수집할 필요는 없다. 하지만 전체 이미지 파일 획득이 불가능하다면 페이지 파일을 따로 수집해야 한다. 또한 페이지 파일은 단서가 될 수 있는 정보가 매우 많이 포함되어 있기 때문에 그에 적합한 분석 방법이 필요하다. 본 절에서는 페이지 파일의 분석 기법을 제시한다.

페이지 파일을 처음부터 4096byte 단위로 구분하였을 때 한 개의 4096byte 페이지의 내용은 모두 동일한 프로세스에서 사용한 내용이다. 4096byte 내의 데이터들을 관련지어서 분석을 한다면 의미 있는 결과가 도출될 가능성이 있다. 또한 페이지 파일에 내용이 기록될 때에는 page-out되는 순서에 따라서 페이지 파일의 offset 1000h부터 4096byte 단위로 순차적으로 기록된다. 이러한 점을 고려해 볼 때, 페이지 파일의 offset 번지에 따라서 page-out되는 시간을 예측할 수 있다. 즉 offset 1000h의 4096byte 페이지보다 그 이후의 1000h 보다 큰 offset부터 시작하는 4096byte 페이지가 시간적으로 뒤에 일어 난 일이라고 분석할 수 있다. 하지만 이러한 분석 내용은 언제나 적용되는 것은 아니다. 왜냐하면 시스템이 page-out된 정보를 페이지 파일의 끝까지 채워서 사용하고, 페이지 파일에 더 이상 채워 넣을 공간이 없다면, 사용을 마치고 반환된 페이지 영역에 그 이후 page-out되는 페이지를 덮어 쓰기 때문이다. 하지만 이러한 시간 순서에 따라 나타나는 분석 결과는 어느 정도 높은 확률로 존재하는데, 그 이유는 시스템을 재실행할 때 마다 페이지 파일의 처음부터 기록하므로 페이지 파일 뒷부분은 덮어 써지는 횟수가 적다. 따라서 페이지 파일의 뒷부분 정보는 시간 순으로 되어 있을 확률이 높다.

3.2.2 데이터 복구

페이지 파일은 텍스트 문자열, 바이너리 코드 값

등이 섞여 있어 분석자가 쉽게 인식할 수 없는 형태이다. 따라서 페이지 파일을 수집하였을 때에는 사람이 읽을 수 있는 형식으로 변환하는 작업이 필요하다.

리눅스의 'strings' 명령은 오브젝트 파일이나 실행 파일에서 문자열을 찾아주는 기능이 있는데 반해 윈도우 유틸리티는 한글 검색을 지원하지 않기 때문에 본 논문에서는 가상 메모리 파일로부터 주요 키워드를 추출하는 필터링 프로그램을 작성하였다. 여러 시스템으로부터 획득한 페이지 파일은 필터링 프로그램을 통해 가독성 있는 형태로 변경한 다음에 분석하였다. 가상 메모리 시스템을 사용하는 이상 페이지 파일에는 "swap-out"된 정보가 하드 디스크에 그대로 남아 있게 되며, 실제로 페이지 파일로부터 다음과 같은 정보들을 획득할 수 있었다.

표 4. 스왑파일(페이지 파일) 내의 정보

1. 문서의 글자
2. ID, 패스워드
3. 사용자가 보았던 내용(URL, 메신저 내용)
4. 타이핑했던 정보
5. 그림파일
6. 기타 정보

대부분의 파일은 자기 자신만의 고유한 시그니쳐를 가지고 있다. 이러한 파일 시그니쳐 검색을 이용한다면 페이지 파일에서 조각난 파일이나 전체를 획득할 수도 있다.

그 예로, JPEG 그림 파일의 시그니쳐를 이용해서 페이지 파일을 검색한 결과 다수의 파일 시그니쳐가 검색됨을 확인하였다. 검색된 해당 바이너리 코드를 hex 편집기를 이용해서 수동으로 복구하였으나, 대부분의 파일이 완전하게 복구되지 않았다. 하지만 이러한 복구 절차가 중요한 이유는 예상치 못한 증거를 획득할 수 있는 기회가 있기 때문이다.

인터넷 포탈사이트 등에서 인터넷 검색을 위해서 검색창에 타이핑했던 정보도 다음과 같이 검출되었다.

표 5. 인터넷 검색 정보

인코딩된 검색 문자열	디코딩 검색어 원문
http://search.naver.com/search.naver?where=nexearch&query=%C0%DA%B5%BF%C2%F7+%C5%B0+%BA%D0%BD%C7	자동차 키 분실

위의 검색 정보의 쿼리 문자열은 url 인코딩[13]되어 있는 정보이기 때문에 url 디코더를 사용해서 다시 원래의 단어를 복원할 수 있고 간단하게 브라우저의 주소창에 전체 url를 붙여넣기만 하더라도 원래의 단어를 알아낼 수 있다. 이러한 정보는 과거에 용의자가 인터넷 검색을 위해서 어떠한 단어를 입력했고 어떠한 정보를 검색하고 있었는가를 판단할 수 있는 중요한 단서를 제공한다.

위의 여러 가지 정보 중에서도 특히 패스워드를 중심으로 조사를 하였는데 메모리에서 볼 수 있었던 사용자 ID와 패스워드가 page-out된 것을 확인할 수 있었다. 그 결과는 다음과 같다.

표 6. 페이지 파일 패스워드 검사 결과

전체 시스템 검사 개수	패스워드 발견 시스템 개수
30개	17개

표 7. 페이지 파일의 크기 별 패스워드 존재 개수

페이지파일 크기	메모리 크기	검사 파일 수	패스워드 존재 수
256M 미만	192M미만	4개	0개
384M	256M	7개	5개
768M	512M	12개	7개
1G 이상	768M이상	7개	5개

표 6을 보면 약 절반 정도에서 패스워드가 존재하는 것을 확인할 수 있으며, 표 7을 보면 페이지 파일의 크기가 384M 바이트 이상일 경우 패스워드가 존재하는 것이 약 65% 이상임을 알 수 있다. 요즘 PC 사양은 메모리 크기가 대부분 256M 바이트

그림 10. jpg 파일 검사 결과

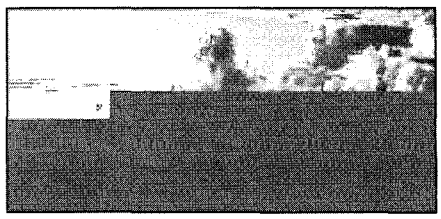


그림 11. 복구된 그림 파일

이상임을 고려할 때 수사에 활용할 수 있는 정보가 더 많이 존재할 것으로 예상된다. 이러한 정보는 범죄 수사 단계에서 용의자의 알리바이 또는 컴퓨터 활용 성향을 알아낼 수 있는 좋은 자료가 될 수 있다.

3.2.3 페이지 파일 분석 의의

메모리와 페이지 파일에는 사용자 입력 정보, ID, 패스워드 등 중요한 개인정보들이 포함되어 있다. 따라서 메모리 정보 수집을 통해 증거수집 직전에 사용하던 용의자의 이메일 또는 웹사이트의 계정과 비밀번호를 쉽게 획득할 수 있다. 또한 페이지 파일 수집을 통해 과거에 기록되었던 개인 정보를 획득할 수도 있다. 만약 범죄 용의자의 시스템인 경우 그의 진술 또는 추가 조사 없이도 이메일 계정의 ID, password, 인터넷 뱅킹 계좌 정보 등을 알 수 있는 장점을 제공한다. 따라서 메모리 분석은 라이브 시스템의 디지털 증거 수집 단계에서 반드시 수행되어야 하는 절차이다. 이러한 메모리 및 페이지 파일 정보 분석을 통한 컴퓨터 포렌식 정보 수집은 컴퓨터 범죄 수사 과정에서 다음과 같은 의미를 가진다.

첫째, 용의자의 중요 정보를 획득할 수 있다. 특히 ID/패스워드는 하드디스크에 저장이 안 되거나, 저장될지라도 암호화 되어 저장된다. 따라서 수사과정에서 ID/패스워드가 필요한 경우 용의자의 진술에 의존하거나, 직접 패스워드를 획득해야 한다. 그러나 메모리 정보 분석 작업은 이를 수사관 스스로 획득할 수 있는 가능성을 제공한다.

둘째, 용의자의 과거 시스템 사용 기록 및 성향을 유추할 수 있다. 페이지 파일에는 과거 메모리 사용 내역이 기록되어 있다. 따라서 이를 분석하면 시스템을 사용한 내역을 유추하여, 수사의 실마리를 제공할 수 있다.

셋째, 하드디스크에서 찾을 수 없는 정보를 제공한다. 용의자가 특정 문서나 그림을 열람하고 이를 Anti-Forensic 기법[10]으로 완벽히 삭제하였을 경우, 이를 하드디스크에서 복구하기가 어렵다. 그러나 문서의 전체 또는 일부의 정보가 메모리에 남은 경우, 이를 복구할 가능성이 있다.

Ⅳ. Window 활성 시스템에 대한 디지털 증거 수집 절차 제안

우리는 앞 절을 통해 메모리 및 가상 메모리 파일을 수집하여 분석한 결과를 제시하였고, 이를 통해 사용자의 중요한 개인정보들을 다수 발견할 수 있음을 확인하였다. 이러한 메모리 및 가상 메모리 파일의 특성을 사용하여 컴퓨터 포렌식 조사/분석 과정에서 용의자의 개인 정보를 획득할 수 있고, 이를 수사 과정의 주요 단서로 사용하거나, 혐의를 입증하거나 반증할 수 있는 디지털 증거로 사용할 수 있다. 이번 절에서는 메모리 및 가상 메모리 파일의 활용을 최대화하기 위해서, 디지털 증거 수집 과정에서 가능한 많은 메모리 정보를 수집할 수 있는 절차를 제안하고자 한다. 제안하는 메모리 정보 획득을 중심으로 한 디지털 증거 수집과정은 다음과 같다.

1. 디지털 증거 수집 현장에 최초로 도착한 자는 용의자의 자료 삭제 및 파괴 행위를 방지한다.
2. 관련 시스템과 증거를 수집할 수 있는 시스템 목록을 작성하고 수색 영장이 허용하는 범위에서 모든 H/W, S/W, 메모, 로그, 주 기억장치, 보조기억장치 등을 수색한다.
3. 작성된 목록에서 휘발성 정보를 포함하고 있는 시스템 및 저장 매체를 식별한다.
4. 메모리 정보를 수집할 수 있는 tool이 포함된 Forensic CD를 사용하여 메모리 덤프를 실시한다. 메모리 정보 수집 tool은 자동으로 정보를 확인 및 수집하는 bat 또는 Shell Script 형태로 구성되는 것이 가장 이상적이다.
5. 획득된 메모리 정보는 용의자의 저장 매체에 저장하지 않고, 디지털 증거 수집자가 준비한 휴대용 저장 매체에 저장하거나, 휴대용 저장 매체를 연결할 수 없을 경우, 네트워크를 통하여 전송하도록 한다.
6. 휘발성 정보를 모두 획득하였다면, 이때 저장 매체의 정보 변경을 최소화 하고 시스템 메모리를 획득하기 위해 최대 절전 모드로 시스템을 종료한다.
7. 부팅이 가능한 컴퓨터 포렌식 CD를 활용하여 재부팅 한다. 컴퓨터 포렌식 CD 내부에 있는 툴을 사용하여, 일반 디지털 증거 수집과 이미징 작업을 실시한다. 만약 필요하다면 pagefile.sys와 hiberfil.sys 파일은 앞 절에서 설명한 방식으로 따로 수집한다. (제안 절차는 메모리 정보 수집을 중심으로 설명하였기 때문에, 이후의 일반적인 디지털 증거 수집 절차는 언급하지 않도록 한다.)

위의 디지털 증거 수집 절차를 사용하면, 최대한

많은 메모리 정보를 수집하여, 향후 디지털 증거 분석 과정에서 용의자에 대한 핵심 정보 획득을 기대할 수 있을 것이다.

V. 결 론

본 연구에서는 RFC3227 증거 수집 지침에 대해서 살펴보았다. 휘발성 데이터를 수집하는 절차에서 정보를 수집하는 것이 좀 더 세부적으로 명시되어 있을 필요가 있다. 따라서 윈도우 시스템의 프로세스 및 전체 메모리 정보와 페이지 파일을 수집하고 그 분석 기법을 제시하였다. 메모리 정보와 페이지 파일에는 개인정보, 그림 파일, 입력한 문자열을 확인할 수 있었고, 특히 패스워드 정보는 약 절반정도 검출되는 것을 확인할 수 있었다.

이는 컴퓨터 범죄 수사 과정에서 용의자를 추적할 수 있는 단서와 정보를 획득할 수 있는 수단으로 사용할 수 있다. 메모리와 페이지 파일에는 수사에 활용할 수 있는 정보가 많이 포함되어 있으므로 반드시 수집을 해야 하며 그에 적합한 분석기법이 필요하다. 또한 시스템 전체 메모리를 획득하고 안전하게 전원을 차단하기 위해서 최대 절전 모드를 사용해야 한다.

향후 연구 과제로는 본 논문에서 제시한 메모리 정보와 페이지 파일의 분석기법을 적용한 포렌식 분석기를 구현하고, 메모리 덤프 파일과 페이지 파일 분석기를 통해 실제적인 증거분석에 대한 실험결과를 도출해 내고자 한다.

참 고 문 헌

[1] RFC3227, "Guidelines for Evidence Collection and Archiving", http://www.faqs.org/rfcs/rfc3227.html, 2002.

[2] Kyle Rankin, "KNOPPIX HACKS", O'RELLY, 2005, pp.256-263

[3] Microsoft Support tools, http://download.microsoft.com/download/win2000srv/Utility/3.0/NT45/E N-US/Oem3sr2.zip

[4] Strings, Sysinternals, http://www.sysinternals.com /Utilities/Strings.html

[5] Virtual Memory, http://searchstorage.techtarget.com/sDefinition/0,,sid5_gci213300,00.html

[6] A. Silberschatz, P. Galvin, "Operating System Concepts", fifth edition, John Wiley & Sons, Inc., 1998.

[7] File Monitor, Sysinternals. http://www.sysinternals.com/ntw2k/source/filemon.shtml

[8] Chris, Kevin, "Incident Response & Computer Forensics", Second Edition, McGraw-Hill, 2003, pp.315-316

[9] Douglas Schweitzer, "Incident Response:Computer Forensics Toolkit", Wiley Pulishing Inc.. 2003, pp. 121-123

[10] 김정민, 박종성, 허재성, 이상진, "Anti-Forensic 기법을 이용한 프라이버시보호에 관한 연구", 한국정보보호학회 하계학술대회 논문집, Vol14, NO1, pp.159-160, june 2004.

[11] ACPO(Association of Chief Police Officers), "Good Practice Guide for Computer based Eletronic Evidence", http://www.acpo.police.uk/asp/policies/p olicieslist.asp, Version 3, pp.11

[12] Chris, Kevin, "Incident Response & Computer Forensics", Second Edition, McGraw-Hill, 2003, pp.114-115

[13] RFC 3875, "The Common Gateway Interface (CGI) Version 1.1", http://www.faqs.org/rfcs/rfc3875.html, 2004

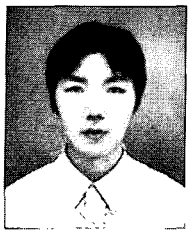
[14] 박종성, 최운호, 문종섭, 손태식, "자동화된 침해사고 대응시스템에서의 네트워 포렌식 정보에 대한 정의", 정보보호학회논문지, 2004년 8월

[15] 이형우, 이상진, 임종인 "컴퓨터 포렌식스 기술", 정보보호학회지, 2002년 10월

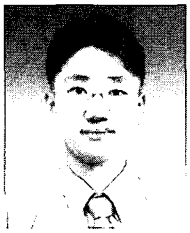
[16] 황현욱, 김민수, 노봉남, 임재명, "컴퓨터 포렌식스: 시스템 포렌식스 동향과 기술", 정보보호학회지, 2003년 8월

[17] 이하영, 김현상, 최운호, 이상진, 임종인, "국내 환경에 맞는 컴퓨터 포렌식에서의 초기 신고 시스템", 정보보호학회지, 2005년 2월

〈著者紹介〉



이 석 희 (Seok-Hee Lee)
2003년: 부경대학교 컴퓨터공학과 졸업(학사)
2004년~2006년: 고려대학교 정보보호 대학원 졸업(석사)
2006년 3월~ : 고려대학교 정보보호 대학원 박사과정
관심분야 : 컴퓨터 포렌식, XML 보안, 역공학



김 현 상 (Hyun-Sang Kim)
2002년: 경희대학교 졸업(학사)
2004년: 고려대학교 정보보호 대학원 졸업(석사)
2005년~현재: 고려대학교 정보보호 대학원 박사과정
관심분야 : 컴퓨터 포렌식, 디지털 증거 인증, 임베디드 포렌식, 분산처리 컴퓨팅

이 상 진 (Sangjin Lee)
1987년 2월: 고려대학교 수학과 학사
1989년 2월: 고려대학교 수학과 석사
1994년 2월: 고려대학교 수학과 박사
1989년 2월~1999년 2월: 한국전자통신연구원 선임 연구원,
1999년 2월~2001년 8월: 고려대학교 자연과학대학 조교수,
2001년 9월~현재: 고려대학교 정보보호대학원 부교수
〈관심분야〉 대칭키 암호, 정보은닉이론, 컴퓨터 포렌식

임 종 인 (Jongin Lim)
1986년 2월: 고려대학교 대학원 수학과 박사(암호학)
2000년 8월: 고려대학교 정보보호대학원/CIST 원장(센터장)
2004년 1월: 국가정보원 정보보호정책자문위원
2005년 7월: 대통령 자문 전자정부 특별위원
2005년 12월: 국회 과기정위원회 정보통신 정책 자문위원
〈관심분야〉 정보보호기술, 정보보호정책, PET, 컴퓨터 포렌식