컴퓨터 포렌식을 위한 한국형 RDS구축

표월성*, 이상진*

*고려대학교 정보보호 대학원

Korea style RDS construction for Computer forensic

Wol-Seong Pyo* ,Sang-Jin Lee*

*Graduate School of information security, Korea University.

요 약

최근 정보통신 및 컴퓨터 산업이 발달함에 따라 해킹 및 바이러스, 산업정보 유출과 같은 컴퓨터를 이용한 범죄가 증가하고 있으며, 범죄에 사용되는 프로그램들이 점차 다양화 되고, 사용되는 데이터의 양 또한 급격히 증가하고 있다. 이에 따라 컴퓨터 범죄를 수사하는 디지털 포렌식 과정에서 시간과 인력 및 비용적인 면에서 방대한 정보들을 모두 분석하기란 현실적으로 어려운 일이다. 그러므로 디지털 증거 분석과정에 앞서 획득한 정보들 중 컴퓨터 범죄와 상관이 없는 정보들을 제거하여 분석할 데이터의 양을 줄이므로 디지털 증거 분석 과정에서 소요되는 인력 및 각종 비용 등을 줄일 수 있다. 이를 위해 국외에서는 이러한 정보들을 웹사이트를 통해 공개 및 제공하고 있다. 그러나 이러한 정보들은 국내에서 발생하는 컴퓨터 범죄를 수사 및 분석하는데 적합하지 않은 경우가 많다. 그러므로 국내의 환경에 맞는 정보를 제공하기 위한 방법 및 제도적인 개선이 필요하다. 본 논문에서는 이러한 정보를 제공하기 위한 방법 및 제도개선 방안을 제안하였다.

I. 서론

일반적으로 컴퓨터 포렌식은 초기 사고 대응 과정(Initial Response) 이후 피해 시스템 및 피해 시스템이 속해 있는 네트워크에서 데이터를 수집(Data Collection)하고, 이를 분석하는 데이터 분석(Data Analysis)과정을 거쳐 형사 및 민사 소송에서 법적인 증거로서 사용될 수 있도록 보고서를 작성하는 일련의 과정을 말한다. 이 중 데이터 수집 과정에서 획득한 정보들에 대한 해시값과 사건과 무관한 정보에 대한 해시값을 비교*하므로 사건과 관련된 파일과 그 외의 파일들을 식별 제거하는 필터링(Filtering) 과정을 거쳐 컴퓨터 포렌식 분석과정에서 소요되는 시간과 비용을 줄일 수 있다.[1] 또한 국내의 정보통신 산업이 발달하면서 인터넷을 사용하는 인구가 증가하고, 이에 따라 다양한 프로그램들이 국외뿐 아니라 국내에서도 많이 개발되어 사용되고 있다. 그러므로 데이터 필터링에 사용할 정보를 획득하기 위해 국내 현실에 맞는 한국형 참조 데이터 셋(RDS, Reference Data Set) 구축 및 제도 개선이 필요하다. 이러한 과정을 통해 보다 효율적이고, 현실적인 컴퓨터 범죄 수사가 가능하다.

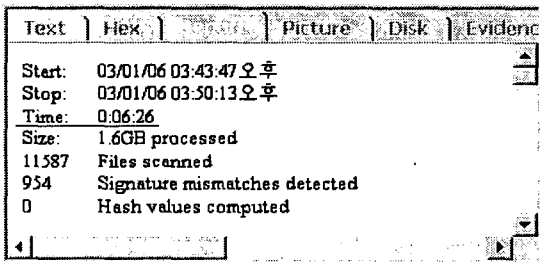
* 본 연구는 정보통신부 대학 IT 연구센터 육성·지원 사업의 연구 결과로 수행되었습니다.

II. 포렌식 분석과정에서 RDS의 필요성

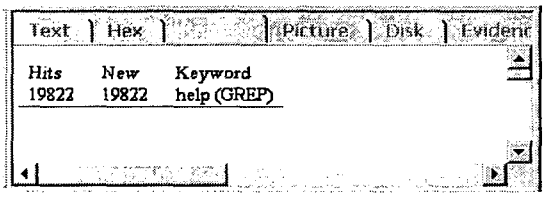
일반적으로 하드디스크에 있는 많은 데이터들은 운영체제와 어플리케이션 파일과 같이 잘 알려진 파일들(Known files)를 포함하고 있다. 이들 잘 알려진 파일들은 사건과 무관한 파일들로 분석 과정에서 이 파일들에 대한 조사는 무의미한 일이다. 그러므로 데이터 획득과정에서 수집한 정보들을 의미 있는 데이터와 그렇지 않은 데이터로 식별하고, 의미 있는 데이터

를 집중 분석하므로 포렌식 분석과정에서 소요되는 시간 및 인력등의 비용을 줄일 수 있을 뿐 아니라 집중된 비용 투자로 인해 효율적인 분석을 할 수 있다. 이를 위해 파일에 대한 무결성을 보장하는데 사용되는 해시 알고리즘을 이용하여 시스템에 있는 모든 파일들에 대한 해시값을 구하고, 분석과정 이전에 보유하고 있던 Hash sets 라이브러리를 서로 비교하므로 사건과 관련 없는 파일들을 빠르게 식별 및 제거하여 불필요한 작업 수행을 막을 수 있다. 이를위해 조사관들은 포렌식 분석과정에 들어가기 전에 잘 알려진 파일들에 대한 해시값들을 보다 많이 획득하여 데이터베이스화 하는 것이 필요하다.

다음은 포렌식 도구로 많이 알려진 EnCase라는 도구를 사용하여, GREP형식의 문자열 "help"에 대한 키워드 검색을 수행한 결과이다.

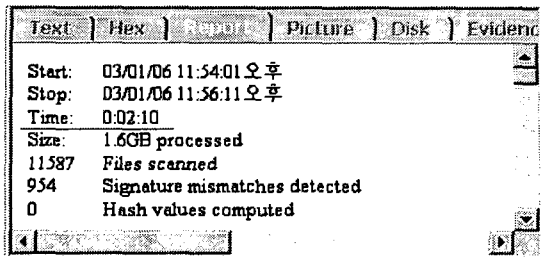


[그림 1] Hash Set을 사용하지 않은 경우의 검색 소요시간

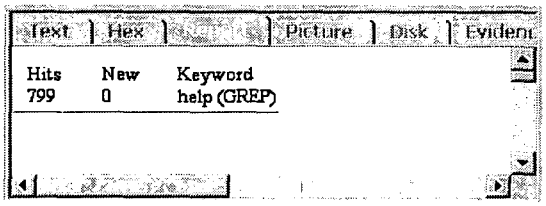


[그림 2] Hash set을 사용하지 않은 경우의 문자열이 검색된 횟수

다음은 Hash set을 사용하여 GREP형식의 문자열 "help"에 대한 키워드 검색을 수행한 결과이다.



[그림 3] Hash Set을 사용한 경우의 검색 소요시간



[그림 4] Hash Set을 사용한 경우의 문자열이 검색된 횟수

위 키워드 검색 결과에서 나타난 것처럼 잘 알려진 파일에 대한 Hash set을 사용하지 않은 경우에 11587개의 파일에서 "help"라는 문자열이 19822번 검색되었으며, 시간은 6분 26초가 소요된 반면에 미리 준비한 Hash set을 사용한 경우 11587개의 파일 중 Hash set에 의해 제외되고 나머지 파일들에서 799번 문자열이 검색되었으며, 시간은 2분 10초가 소요되었다. 이처럼 잘 알려진 파일에 대하여 미리 획득한 Hash set을 이용하므로 분석과정에서 소요되는 비용을 줄일 수 있었다. 그러므로 가능한 많은 양의 잘 알려진 파일들에 대한 RDS를 구축하는 것이 필요하다.

Ⅲ. 현재 참조 데이터 셋 제공 현황

3.1 국외의 현황

국외에서는 데이터 필터링을 위해 컴퓨터 범죄와 관련이 없는 각종 운영체제 및 어플리케이션 파일들에 대한 MD5, SHA-1과 같은 해시값을 제공하고 있다. 또한 해킹 및 컴퓨터 범죄에 사용되고 있는 익스플로잇과 같은 파일에 대한 정보를 컴퓨터 포렌식 뿐 아니라 여러 산

업에서 이용할 수 있도록 웹사이트를 통해 제공하고 있다.

■ NIST NSRL RDS(Reference Data Set)

참조 데이터 셋 정보(이하 RDS)를 제공하는 가장 대표적인 웹사이트로써, 미국표준기술연구소(NIST, National Institute of Standards and Technology)의 NSRL(National Software Reference Library)프로젝트를 통하여 컴퓨터 포렌식 분야 및 법원, 정부, 일반 산업체에 OS, 시스템 파일과 같은 잘 알려진 파일과 다양한 어플리케이션등의 Digital Signature를 제공하고 있다. 이 사이트 에서는 버전 2.1로 부터 시작하여 현재 버전 2.11까지 업데이트 되었으며, 4개의 ISO 이미지 파일과, MD5 및 SHA-1값만을 모아놓은 2개의 Zip파일을 배포하고 있다.[2]

■ Known goods

여러 버전의 리눅스 및 유닉스 시스템에 대한 MD5, SHA-1 해시값들을 검색 기능을 통해 제공하고 있다.[3]

■ CyberAbuse Rootkit ID 프로젝트

다른 사이트와는 달리 유닉스 계열의 루트킷(Rootkit)파일들에 대한 SHA1 값을 데이터베이스화 하여 소프트웨어를 통해 탐지할 수 있도록 하고 있다. 그러나 현재는 업데이트가 중단된 상태이다.[4]

■ Rootkit Hunter 프로젝트

유닉스 계열의 루트킷(rootkits), 백도어(backdoors), 익스플로잇(exploits)에 대한 정보를 데이터베이스화 하고, Rootkit Hunter란 도구를 사용하여 검색할 수 있도록 하고 있다. 지원하는 운영체제는 대부분의 Linux 와 BSD계열을 지원하며, Bourne Again Shell (BASH)만을 지원하고 있다.[5]

3.2 국내 현황

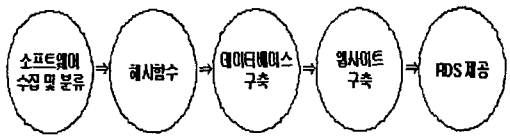
현재 국내에서는 미국 NIST의 NSRL프로젝트와 같이 공개적으로 참조 데이터 셋(RDS)을 제공하는 웹사이트 및 단체가 전무한 형편이다.

Ⅳ. 한국형 K-RDS 구축 및 활용

국외에서 제공하는 RDS에서는 국내에서 개발되거나 사용되고 있는 한국어 버전의 운영체제 및 장치 드라이버, 특히 아래 한글과 같은 오피스관련 어플리케이션과 백신 프로그램등에 대한 RDS정보는 포함하지 않고 있다. 그러므로 효율적인 컴퓨터 포렌식 분석과정을 위해 국내 현실에 맞는 RDS를 제공하는 것이 필요하다.

4.1 한국형 RDS 구축 과정

한국형 RDS를 구축하는 방법은 아래와 같이 5단계로 나뉜다.



[그림 5] 한국형 RDS 구축 과정

① 소프트웨어 수집 및 분류 단계

국내에서 개발된 소프트웨어 및 소스등을 수집하여 각 용도와 목적에 맞게 분리하는 단계이다.

② 해시 함수 단계

①단계에서 수집한 소프트웨어에 대하여 국내 해시 알고리즘(HAS160)를 적용하여 해시값을 생성하는 단계이다. 또한 이 단계에서는 NIST의 NSRL RDS와 호환을 위해 MD5, SHA-1값을 생성을 고려하였다.

③ 데이터베이스 구축 단계

이전 단계에서 획득한 해시값을 배포를 위해 파일로 생성하여 저장하는 단계이다.

④ 웹사이트 구축 단계

K-RDS 제공을 위해 고려대학교 정보보호대학원 포렌식 연구실의 웹사이트에 해당 카테고리를 구축하는 단계이다.

⑤ RDS 제공 단계

웹사이트를 통해 파일을 배포하는 단계이다. 배포할 파일은 전체 5개이며, 각 파일에 대한 필드 정보는 다음과 같다.

■ KHashes.txt

배포할 파일들에 대한 해시값을 저장한 파일

데이터형식	"FileName","has-160"	
필드이름	FileName	파일이름
	has-160	파일이름에 대한 has160 해시값
예	"KHFile.txt","75d7d09633d573e6f6a90a52b5f803b5af065fd2"	

[표 1] KHashes.txt파일 데이터형식

■ KMfg.txt

국내 제조사 정보를 저장한 파일

데이터형식	"MfgCode","MfgName"	
필드이름	MfgCode	제조사 식별을 위하여 회사의 영문이름을 일부 혹은 전체를 사용하여 표시
	MfgName	회사의 전체 이름
예	"NCsoft","NCsoft Corporation"	

[표 2] KMfg.tx파일 데이터형식

■ KOSinfo.txt

운영체제 정보를 저장한 파일

데이터형식	"OsCode","OsName","OsVersion","MfgCode"	
필드이름	OsCode	운영체제 식별번호/문자
	OsName	운영체제 이름
	OsVersion	운영체제 버전정보
	MfgCode	제조사 식별번호/문자
예	"Win2kPro","Windows 2000 Professional","Unknown","Microsoft"	

[표 3] KOSinfo.txt파일 데이터형식

■ KProducts.txt

소프트웨어에 대한 정보를 저장한 파일

데이터형식	"ProductCode","ProductName","ProductVersion","OsCode","MfgCode","Language","SoftwareType"	
필드이름	ProductCode	소프트웨어 식별번호
	ProductName	소프트웨어 이름
	ProductVersion	소프트웨어 버전
	OsCode	운영체제 식별코드
	MfgCode	제조사 식별번호/문자
	Language	사용언어
	SoftwareType	소프트웨어 타입
예	"10000","Alzip","6.21","Win2kPro","ESTsoft","korean","utility"	

[표 4] KProducts.txt파일 데이터 형식

■ KHFile.txt

제공할 RDS 정보로써 파일에 대한 해시값을 저장한 파일

데이터형식	"has-160","CRC32","FileName","FileSize","ProductCode","OsCode","FileSignature"	
필드이름	has-160	파일에 대한 has160 해시값
	CRC32	crc32값
	FileName	파일의 이름
	FileSize	파일의 크기
	ProductCode	소프트웨어 식별번호
	OsCode	운영체제 식별코드
	FileSignature	
예	"37667cb9f6752783e1423e4ba3cedf6336a4e148","5ECA212F","인터넷 연결 마법사.lnk",728,2222,"Win2kPro",""	

[표 5] KHFile.txt파일 데이터형식

V. 발전된 한국형 RDS 구축을 위한 제도적 개선 방법

국내 현실에 맞는 RDS를 생성하기 위해 국내에서 개발된 모든 소프트웨어들을 개인이나 소규모 단체에서 수집 및 관리하는 것은 인력 및 시간, 비용 등을 고려해 볼 때 현실적으로 불가능한 일이다. 이렇게 소요되는 각종 비용들을 줄이기 위해 본 논문에서는 현재 실행되고 있는 기존의 제도적 절차를 검토하고, 효율적으

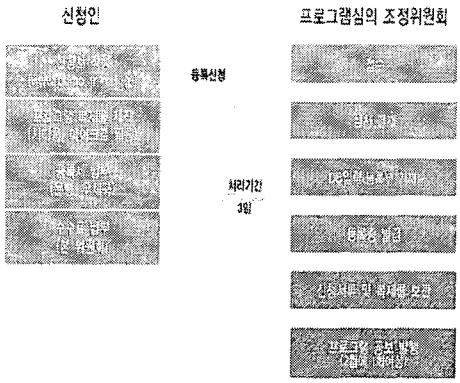
로 RDS를 생성하기 위한 국가 차원의 제도 개선 방안을 제시하였다.

5.1 현재 실행중인 제도적인 절차

보통 기업에서 소프트웨어를 개발한 이후, 개발한 소프트웨어에 대하여 창작 및 소유권을 법적으로 명확히 하여 이에 대한 권리를 보호받기를 원한다. 이를 위해 지정된 위탁관리 기관을 통해 해당 소프트웨어를 등록한다. 소프트웨어 등록을 수행하는 기관 및 과정은 다음과 같다.

■ 프로그램심의 조정위원회

현재 국내에서 개발된 소프트웨어는 컴퓨터프로그램보호법 제20조(프로그램저작권 위탁 관리기관 지정 등) 및 제27조(업무의 위탁)에 관한 법률[6]에 의하여 프로그램심의조정위원회에서 프로그램 등록업무를 위탁 수행하고 있으며, 전체적인 프로그램 등록 절차는 아래의 그림과 같다.

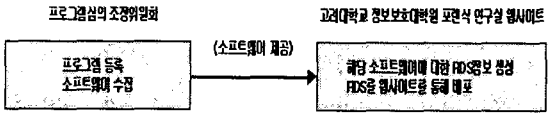


[그림 6] 프로그램 등록 과정

위 그림처럼 프로그램을 개발한 신청인이 프로그램등록 신청서, 프로그램복제물, 등록세 납부 영수증 등을 구비하여 등록을 신청하고, 프로그램심의 조정위원회에서는 심사를 통해 등록증을 발급하고 있다.[7]

5.2 제도 개선 방안

기존에 존재하는 프로그램 등록과정 중에서 컴퓨터프로그램보호법 제24조(프로그램 제출)와 컴퓨터프로그램보호법시행령 제18조(프로그램복제물의 제출)에 근거하여 소프트웨어에 대한 복제물을 접수 및 처리하고 있다. 이를 통해 RDS를 생성하기 위해 필요한 소프트웨어를 개인 및 소규모 단체에서 따로 수집할 필요가 없으며, 이로 인해 수집에 소요되는 상당한 비용을 줄일 수 있다.



[그림 7] 상호 연계 과정

각 기관의 상호 연계과정을 통한 제도개선을 하므로 다음과 같은 몇 가지 장점이 있다. 첫째, 역할 분담을 통해 효율적인 RDS정보 획득. 둘째, 소프트웨어 수집 과정에서 발생하는 비용 등을 줄일 수 있다. 셋째, 프로그램심의 조정위원회에서 프로그램의 창작 및 권리변동 사항을 처리하고 있으므로 소프트웨어에 대한 RDS배포 시 만약 있을 수 있는 기업의 기술유출 및 저작권에 관한 분쟁을 막을 수 있다

VI. 결론

본 논문에서는 컴퓨터 포렌식 과정에서 컴퓨터 범죄와 관련이 없는 파일들을 제거하는 필터링 기능을 통해 포렌식 분석 과정에서 소요되는 각종 비용을 줄일 수 있다. 그러므로 분석 과정에 들어가기 전에 필터링에 사용되는 데이터 정보를 확보해야 한다. 이를 위해 국외에서는 웹사이트를 통해 RDS정보를 제공하고 있으나 현재 국내에서는 특별히 데이터 정보를 제공하는 공간이 없다. 또한 인터넷의 발달로 인해 국내에서 수많은 각종 소프트웨어가 개발되고 있으며, 국내에서 벌어지는 컴퓨터 관련 범

최에 사용되기도 한다. 그러므로 국내 환경에 맞는 RDS를 생성하여 제공하는 것이 필요하다. 또한 현재 실행중인 제도를 이용하는 제도 개선을 통해 프로그램심의 조정위원회의 소프트웨어 등록과정을 이용하여 소프트웨어를 수집하고, 고려대학교 정보보호대학원의 포렌식 연구실과 연계하여 RDS생성 및 배포를 하므로 RDS를 생성과정에서 소요되는 비용을 줄일 수 있다.

[참고문헌]

[1] Chris Prosise, Kevin Mandia, "Incident Response & Computer Forensics, Second Edition", p.282, McGraw-Hill Companies 2003

[2] National Institute of Standards and Technology, "descriptin of the RDS contents", http://xsun.sdct.itl.nist.gov/~dwhite/RDS/rds_2.11/read_me.txt", NSRL project 2005

[3] known goods "KnownGoods Database", http://www.knowngoods.org/

[4] Philippe Bourcier, Stephane Thiell "The CyberAbuse Rootk(it)ID Project", http://rk.cyberabuse.org/, RootkID 2002.

[5] Michael Boelen, "Rootkit Hunter", http://www.rootkit.nl/projects/rootkit_hunter.html, Rootkit Hunter Projects 2006.

[6] 법제처, "컴퓨터프로그램보호법 2006년 7월1일시행", http://www.klaw.go.kr/CNT2/Easy/MCNT2EasyLawService.jsp?s_lawmst=72350, 법제처 종합법령정보센터, 2005

[7] 프로그램심의 조정위원회, "프로그램 등록업무", http://www.pdmc.or.kr/affairs/registration/intro.jsp