분석 목적별 분류기반의 데이타베이스 포렌식 모델

(A Database Forensics Model based on Classification by Analysis Purposes)

김 성 혜 † 김 장 원 †† 조 은 애 †† 백 두 권 †††
(Sunghye Kim) (Jangwon Kim) (Eun-Ae Cho) (Doo-Kwon Baik)

요 약 디지털 포렌식이란 범죄와 관련된 전자적 증거를 찾아내는 것이다. 사이버 범죄가 날로 증가하는 가운데 전자적 증거를 찾아내는 디지털 포렌식 연구의 중요성도 부각되고 있다. 현재 디지털 포렌식 분야는 디지털 포렌식을 위한 전반적인 프로세스 모델과 디스크 포렌식, 네트워크 포렌식, 시스템 포렌식, 데이타베이스 포렌식과 같은 분야별 분석기법에 대한 연구가 진행되고 있다. 지금까지 데이타베이스 포렌식 분야에서는 특정 벤더에 종속적인 분석기법을 제안하기 때문에 다양한 데이타베이스에서 일반적으로 사용할 수 있는 프로세스 모델 및 분석기법에 대한 연구를 진행되지 않고 있다. 이 논문은 디지털 포렌식 분야 중 데이타베이스 포렌식을 위한 논문으로 데이타베이스 포렌식의 프로세스 모델과 분석기법을 통합한 모델을 제안한다. 제안하는 데이타베이스 포렌식 모델(DFM)은 기존 문제를 해결하고 각 상황과 목적에 따라 데이타베이스를 분석 할 수 있으며 다양하게 발생할 수 있는 데이타베이스 분석에서 정형화된 모델과 분석기법을 사용 할 수 있다. 실험은 다양한 데이타베이스 분석에서 제안하는 DFM을 실제 적용함으로써 현장에서 증거 수집뿐만 아니라 데이타들의 관계를 분석하는 것까지 적용 가능한 것을 확인 할 수 있다.

키워드 : 데이타베이스 포렌식, 디지털 포렌식, 데이타베이스 보안

Abstract Digital forensics refers to finding electronic evidences related to crimes. As cyber crimes are increasing daily, digital forensics for finding electronic evidences is also becoming important. At present, various aspects of digital forensics have being researched including the overall process model and analysis techniques such as network forensics, system forensics and database forensics for digital forensics. Regarding database forensics, only analysis techniques dependent on specific vendors have been suggested. And general process models and analysis techniques which can be used in various databases have not been studied. This paper proposes an integrated process model and analysis technique for database forensics. The proposed database forensics model (DFM) allows us to solve problems and analyze databases according to the situation and purpose, and to use a standard model and techniques for various database analyses. In order to test our model(DFM), we applied it to various database analyses. And we confirmed the results of our experiment that it can be applicable to acquisition in the scene as well as analysis of data relationships.

Key words : Database Forensics, Digital Forensics, Database Security

· 이 논문은 2008 한국컴퓨터종합학술대회에서 '분석 목적별 분류 기반의 데이타베이스 포렌식 모델'의 제목으로 발표된 논문을 확장한 것임

논문접수 : 2008년 8월 25일
심사완료 : 2008년 11월 10일

† 정 회 원 : 고려대학교 소프트웨어공학과
kimsh96@korea.com
†† 학생회원 : 고려대학교 컴퓨터 · 전파통신공학과
ikaros1223@korea.ac.kr
eacho@korea.ac.kr
††† 종신회원 : 고려대학교 컴퓨터 · 전파통신공학과 교수
baikdk@korea.ac.kr

정보과학회논문지 : 데이타베이스 제36권 제2호(2009.4)

1. 서 론

컴퓨터의 보급이 일반화 되고 사이버 범죄가 급증하면서 이로 인해 디지털 포렌식의 중요성이 점점 부각되고 있다.

컴퓨터 포렌식(Computer Forensics)은 IACIS(International Association of Computer Specialists)에서 처음 제안된 것으로, 컴퓨터를 기반으로 한 범죄의 전자적 증거(electric evidence) 조사와 관련한 연구를 하는 것이다[1]. 현재는 컴퓨터 분야에만 국한되지 않고 디지털 증거에 대해 좀 더 포괄적인 개념으로서의 디지털 포렌식(Digital Forensics)이란 의미로 사용되고 있다.

디지털 포렌식은 디지털 포렌식 분야별 분석기법과 디지털 포렌식 전반적인 프로세스 모델에 대한 연구로 나뉜다.

먼저, 디지털 포렌식 분야에는 디스크 포렌식(Disk Forensics), 운영체제에 따른 시스템 포렌식(System Forensics), 인터넷 포렌식(Internet Forensics), 네트워크 포렌식(Network Forensics), 데이타베이스 포렌식(Database Forensics), 모바일 포렌식(Mobile Forensics)으로 세분화되어 연구되고 있다[2-5].

디스크 포렌식은 물리적인 저장 매체인 하드 디스크, 플로피 디스크, 각종 보조기억 장치에서 증거를 수집하고 분석하는 것으로 일반적으로 가장 많이 연구되고 있는 분야이다. 시스템 포렌식은 컴퓨터 시스템의 운영체제, 서비스, 응용프로그램 및 프로세스를 분석하는 것이다. 네트워크 포렌식은 네트워크를 통하여 전송되는 데이타나 패스워드 등 데이타의 트래픽을 분석하거나, 접근·에러로그, 네트워크 환경을 조사하고 분석한다. 데이타베이스 포렌식은 데이타베이스와 관련된 전반적인 분석으로 분석 대상을 RDBMS에 적용하여 복원하고 분석하는 절차 및 방법이다[5]. 데이타베이스를 이용한 분석으로 전산자료나 데이타파일을 통해 데이타를 추출하고 분석한다. 모바일 포렌식은 PDA, 휴대폰, MP3, 전자수첩, 네비게이션 등 휴대기기의 정보를 분석한다.

분야별로 분석기법에 대한 연구가 활발하게 진행되는 가운데 데이타베이스 포렌식에 대한 관심도 점차 증가하고 있다. 그러나 디지털 증거를 분석할 때 일반적으로 사용하는 도구인 Guidance사의 Encase[6], AccessData사의 FTK[7]는 데이타베이스의 중요한 특징 중 하나인 데이타들의 관계를 분석할 수 없는 문제가 있다. 또한 데이타베이스 포렌식의 연구가 특정 제품에 국한된 연구가[8-13] 주로 이뤄지고 있기 때문에 특정 데이타베이스의 물리적인 구조에 종속되어 물리적 구조가 바뀔 경우 분석기법을 재사용할 수 없는 문제를 가진다.

다음으로 디지털 포렌식 전반적인 프로세스 모델은 증거를 수집하고 분석하는 디지털 포렌식 표준화 및 디지털 조사 프로세스에 대한 연구가 있다[14-17]. 그러나 데이타베이스 포렌식 분석을 위한 전반적인 정형화 모델과 분석기법이 없는 문제점이 있다. 따라서 데이타베이스 포렌식을 위한 정형화된 모델과 분석기법을 통합한 연구가 필요하게 되었다.

기존의 문제점을 해결하기 위해, 본 논문에서는 분석 목적별 분류에 따른 데이타베이스 포렌식 모델(Database Forensic Model: DFM)을 제안한다. 제안하는 모델은 다양한 데이타베이스를 먼저 분석 목적에 따라 네 가지 모델로 분류하고, 그 다음 각 모델별 분석기법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 디지털 포렌식의 전반적인 프로세스 모델인 디지털 포렌식 연구를 위한 로드맵[14]과 일반적인 프로세스 모델(Common Process Model)[15]을 살펴보고 데이타베이스 포렌식과 관련된 분석기법 통해 기존 연구의 문제점을 살펴본다. 3장에서는 분석 대상 목적에 따른 네 가지 모델과 18가지 분석기법에 대해 설명한다. 제4장에서는 제안하는 분석모델과 분석기법을 실제 적용하여 검증을 한다. 5장에서는 결론 및 향후 연구에 대해 제시한다.

2. 관련 연구

현재 디지털 포렌식의 전반적인 프로세스에 대한 연구는 디지털 포렌식 연구를 위한 로드맵[14]과 일반적인 프로세스 모델[15] 등이 있으며 데이타베이스 포렌식 분석에 대한 연구는 데이타베이스 시스템의 포렌식 분석[18]와 컴퓨터 포렌식 가이드라인[19], 오라클 포렌식[8-12], MS-SQL 서버 포렌식[13] 등이 있다.

2.1 디지털 포렌식의 전반적인 모델

디지털 포렌식의 전반적인 모델에 대한 연구는 DFRW (Digital Forensic Research Workshop)에서 발표한 디지털 포렌식 연구를 위한 로드맵[14]과 일반적인 프로세스 모델[15] 등이 있다.

먼저 DFRW에서 연구한 방법은 디지털 포렌식의 유형을 네 가지로 분야로 정의 한다. 네 가지 분야는 첫 번째 디지털 포렌식의 특징에 대한 연구, 두 번째 디지털 증거의 무결성에 대한 연구, 세 번째 삭제된 데이타 복구에 대한 연구, 네 번째 네트워크 포렌식에 대한 연구이다. 그러나 데이타베이스 포렌식 분야는 포함되지 않는다. 일반적인 파일 시스템을 이용한 복원 방법으로는 데이타들의 관계를 통한 데이타를 추출할 수 없는 문제점이 있기 때문에 별도의 데이타베이스 포렌식 분야로 분리되어 연구되어야 하지만 DFRW에서는 다루지 않고 있다.

기존의 디지털 포렌식 모델은 현장의 사고 대응

(Incident Response: IR)과 디지털 포렌식(Digital Forensics: DF)으로 나누고 있었다. 그러나 일반적인 프로세스 모델은 IR과 DF로 나누지 않고 사고감지(Pre-Incident Preparation), 분석 전(Pre-Analysis Phase), 분석(Analysis Phase), 기록(Post-Analysis)으로 여러 개의 단계를 구성된 모델을 제시하기 때문에 데이타베이스 시스템을 적용하기에는 포괄적인 한계를 가지고 있다.

2.2 데이타베이스 포렌식 분석 연구

데이타베이스 포렌식 분석에 대한 연구는 데이타베이스 시스템의 포렌식 분석, 컴퓨터 포렌식 가이드라인 중 데이타베이스 포렌식, 오라클 포렌식, MS-SQL 서버 포렌식 등이 있다.

먼저 데이타베이스 시스템의 포렌식 분석은 데이타베이스에서 삭제하였으나 DB Slack이나 FS Slack에 저장된 데이타를 복구하여 분석한다.

그림 1은 데이타 생명주기(Lifetime)[18]의 흐름도이다.

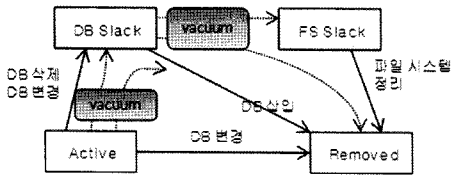

그림 1 데이타 생명주기(Lifetime)

데이타 생명주기(Lifetime)에 따르면 데이타가 삭제되었지만 실제 DB Slack에 남아 있어 삽입(Insert) 되어 덮어쓰기(overwrite) 전까지는 DB Slack에 저장되어 있기 때문에 데이타를 복구할 수 있다.

그러나 데이타베이스에서 특정 대상을 추출(select)하거나, 관계(relation)를 통한 분석, 현장에서의 데이타 수집과 같은 여러 가지 분석상황에 적용할 수 없는 문제점을 가지고 있다.

컴퓨터 포렌식 가이드라인은 디지털 포렌식을 분석기법에 따라 분류하고 절차를 제시한다. 디지털 포렌식의 분석기법은 디지털 증거물 절차, 데이타베이스 분석 절차, 악성코드 분석 절차, 암호 해독 절차, 전자우편 분석 절차로 기술한다. 제시하는 데이타베이스의 분석 절차는 분석 대상에 대한 분류 기준 없이 분석 항목별 판단으로 절차화 되어 분석 대상과 목적과 관계없이 분석관의 판단에 따라 분석기법과 결과가 달라질 수 있는 문제점을 가지고 있다.

오라클 포렌식은 Redo Log, Dropped Object, Undo Segment등과 같이 오라클의 데이타파일이나 로그파일에 대한 분석기법을 제안한다. MS-SQL 서버 포렌식은 MS-SQL 서버의 분석 로그파일이나 데이타파일 분석을 제안한다. 오라클 포렌식과 MS-SQL 서버 포렌식은 특정 제품에 종속적이며 데이타들의 관계(relation)를 간과하는 문제를 가지고 있다.

지금까지 살펴본 결과 디지털 포렌식의 전반적인 프로세스 모델에서 데이타베이스 포렌식도 네트워크 포렌식과 디스크 포렌식처럼 별도 연구가 필요하지만 현재는 활발히 연구되고 있지 않는 상태이다. 그리고 데이타베이스 포렌식의 분석기법은 특정 벤더에 종속적이거나, 데이타베이스의 여러 분석기법 중 단편적인 분석기법에 대한 연구만 이루어지고 있다.

그러므로 데이타베이스 포렌식의 전반적인 프로세스 모델에서부터 분석기법까지, 데이타베이스의 벤더에 종속되지 않고 상황별로 분석대상의 목적에 따라 적용시킬 수 있는 새로운 데이타베이스 포렌식 분석 방법론이 필요하다.

3. 데이타베이스 포렌식 모델

3.1 데이타베이스 포렌식 목적별 분류

제안하는 기법은 분석 대상을 첫 번째 단계로 목적으로 분류를 하고 두 번째 단계로 분류된 모델의 분석 절차를 제시함으로써 다양한 데이타베이스를 하나의 제안 모델을 적용하여 효율적으로 분석할 수 있다.

본 연구는 데이타베이스의 분석 목적을 다음과 같이 네 가지로 분류한다.

• DBMS 분석

DBMS 분석은 분석 목적이 데이타베이스 시스템(Database System)을 대상으로 하는 경우이다. 데이타베이스 서버가 해킹을 당하거나 데이타가 유출 되었을 경우, 그리고 데이타베이스에서 특정 대상의 데이타를 추출해야 하는 경우뿐 아니라, 백업된 데이타를 복원하여 분석 할 수 있다. DBMS 분석은 직접 분석과 간접 분석으로 나눌 수 있다. 먼저 직접 분석은 데이타베이스 원본을 직접 모니터링하거나 데이타를 추출하는 것을 말한다. 그리고 간접 분석은 데이타베이스 원본을 직접 분석하지 않고 별도 분석용 데이타베이스 서버를 구축하여 데이타베이스 원본에서 백업 파일과 같은 데이타베이스 파일을 추출하여 분석용 데이타베이스 서버에 복원한 뒤 데이타의 관계를 분석하고 추출하는 간접 분석이 있다. DBMS 분석은 데이타베이스의 데이타들의 관계(relation)를 통해 데이타를 추출하는 경영자료, 회계 자료와 같은 간접 분석으로 많이 사용한다.

• 소스를 통한 분석

소스를 통한 분석은 분석 목적이 데이타베이스 시스템뿐만 아니라 웹사이트 혹은 애플리케이션 소스를 동시에 분석하는 경우이다.

데이타베이스 포렌식에서는 '데이타를 어떻게 가공하

느냐'가 중요한 문제인데 소스분석을 통해 데이타 간의 관계를 분석뿐만 아니라 데이타베이스의 위치와 종류를 파악할 수 있다. 그리고 데이타베이스에 저장되지 않은 데이타의 정보를 확인할 수 있다.

초기에는 데이타베이스 서버만 압수해서 오는 경우가 많았으나 데이타베이스 서버만으로는 명확하게 데이타를 추출하는 데에는 한계가 있었기 때문에 현재는 소스를 동시에 압수한다.

• 데이타파일 분석

데이타파일 분석은 분석 목적이 데이타베이스 파일을 직접 분석 하는 방법으로, 손상되거나 삭제된 데이타파일을 데이타베이스 시스템에 복원하지 않고 hex프로그램을 이용하여 파일에 저장된 값을 분석한다.

파일의 시작(header)이나 끝(footer)이 손상된 데이타파일을 경우 정상적으로 데이타베이스에 복원되지는 않지만, 파일에는 내용이 저장되어 있다. 데이타베이스 파일이 시작(header) 검색을 통해 삭제된 파일의 위치를 찾을 수 있다. 뿐만 아니라 앞서 본 데이타베이스 시스템의 포렌식 분석[18]과 같이 삭제된 데이타도 파일 Slack에 저장 되어 있으므로 덮어쓰기 전 까지는 삭제된 데이타들도 추출할 수 있다.

• 현장에서의 데이타 수집 및 분석

현장에서의 데이타 수집 및 분석은 현장에서 데이타를 수집하고 분석하는 경우이다. 현장에는 여러 대의 데이타서버가 존재하므로 필요한 데이타가 저장된 데이타베이스 서버를 식별하는 것이 중요하다. 현장에서 서버를 압수할 수 있지만 필요한 데이타의 내용만 데이타베이스 백업한다. 그 후 압수한 데이타베이스 백업을 분석용 데이타베이스 서버에 복원하여 DBMS 분석기법을 통해 분석한다.

또한 현장에서는 모니터링을 통해 접속 기록이나 작업 내용도 확인할 수 있다.

현장에서 데이타베이스를 백업하거나, 로그 파일을 수집하고 난 후 압수 파일의 파일 목록과, 해시값(hash), 시스템 정보를 기록하는 [체크리스트]를 작성한다.

3.2 데이타베이스의 목적별 분류에 따른 분석 절차

이 절에서는 데이타베이스의 분석 목적에 따라 데이타베이스 포렌식 모델에 따른 분석기법을 정의한다.

분석기법은 분석을 수행하는 각각의 분석 처리 단위이므로 프로세스로 명명한다.

제안하는 데이타베이스 포렌식 모델에서는 표 1과 같이 18개의 프로세스를 정의한다.

각 프로세스에 고유한 번호를 부여하여 분석 대상의 목적에 따라 프로세스를 선택 및 배열할 수 있다. 이것은 추후에 분석 목적을 추가하거나 기존의 분류에 프로세스의 추가를 가능하게 한다.

표 1 프로세스 번호(PID)별 프로세스 명

PID	프로세스 명	PID	프로세스 명
1	DB 파일 위치확인	10	소스 파일 분석
2	시스템 파일 분석	11	파일 헤더 검색
3	파일 복원	12	키워드 검색
4	복원	13	데이타파일 분석
5	DB 시스템 확인	14	DB 수집 대상 선별
6	데이타 분석	15	데이타 백업
7	로그 분석	16	모니터링 수집
8	소스 디렉토리 확인	17	수집 대상 파일
9	공통 참조 파일 분석	18	체크리스트 작성

각 프로세스의 역할은 다음과 같다.;

(1) DB 파일 위치확인

쓰기 방지 장치(Write Blocker Device)를 원본 또는 사본에 연결하여 데이타베이스와 관련된 데이타파일(Datafile)이나 설정(config)파일, 로그파일(Log File)의 위치를 확인한다.

(2) 시스템 파일 분석

DB파일 위치를 확인하고, 데이타베이스와 운영체제에서 기록하는 시스템 파일을 분석함으로써 서버의 접속기록 및 데이타베이스 접속기록을 확인한다. 그리고 데이타 백업위치, 백업모델을 파악한다.

(3) 파일 복원

데이타베이스 분석에 필요한 파일을 별도로 저장(copy)한다. 앞서 설명한 직접 분석을 할 경우에는 데이타베이스 서버에서 수집한 내용을 별도 저장하고, 간접 분석의 경우는 분석용 데이타베이스 서버에 복원(recovery, restore)하기 위해 필요한 파일들을 별도의 저장 매체에 저장(copy)한다.

(4) 복원

복원은 데이타들의 관계(relation)를 통해 필요한 데이타를 추출하기 위해 분석용 데이타베이스 서버에 원본 데이타베이스 파일들을 저장(copy)하고 복원(recovery, restore)하고 시작(startup)한다.

(5) DB시스템 확인

데이타베이스를 복원한 다음 데이타베이스에 대한 시스템 정보를 확인한다. 데이타베이스의 버전과 스케줄러, 프로시저, 함수, 테이블 설명(comment)을 확인한다.

(6) 데이타 분석

DB 시스템에서 확인된 데이타베이스 정보와 프로그램 소스가 있을 경우는 소스와 조합하여 데이타들의 관계를 분석하여 데이타를 추출한다.

(7) 로그 분석

데이타베이스의 로그에는 데이타를 복원하기 위해 변경된 데이타의 변경 전 데이타와 변경 후 데이타가 저장되어 있다. 로그 분석을 통해 데이타가 언제, 어떤 데

이타로 변경 되었는지 확인한다.

(8) 소스 디렉토리 확인

프로그램 소스를 분석 할 때 먼저 소스의 폴더 구조를 확인한다. 소스는 동일한 특성을 가진 파일들끼리 동일한 폴더에 저장하기 때문에 반드시 확인한다.

(9) 공통 참조 파일 분석

소스 디렉토리를 통해 프로그램들이 공통적으로 사용하는 데이타베이스 접속 파일이나 데이타 정보 파일은 별도로 저장한다. 일반적으로 Include, Function 파일이나 폴더에 저장되어 있다.

(10) 소스 파일 분석

응용 프로그램 소스 파일을 분석하여 데이타간의 관계를 명확하게 분석할 수 있다.

(11) 파일 헤더 검색

삭제된 데이타파일을 복구하기 위해서 데이타파일이나 백업 파일의 헤더(header)를 검색해서 데이타파일의 일부분을 찾는다.

(12) 키워드 검색

데이타의 속성을 키워드로 수식이나 명칭으로 검색하여 파일의 Slack이나 손상된 데이타파일에서 데이타를 추출한다.

(13) 데이타파일 분석

데이타파일은 특정 제품마다 물리적으로 다르게 저장된다. 데이타파일 분석은 물리적 구조에 종속적으로 제품마다 다르게 분석 하여야 한다. 앞에서 살펴본 오라클 포렌식, MS-SQL 서버 포렌식을 이용한다.

(14) DB 수집 대상 선별

현장에서 데이타베이스 압수 시 여러 대의 데이타베이스 서버가 연동하여 동작 할 때 데이타베이스 수집 대상을 우선적으로 선별한다.

(15) 데이타 백업

데이타베이스 서버를 물리적으로 분리하는 것이 아니라 일정 범위의 대상만 추출하여 데이타를 백업 하거나, 데이타베이스 온라인 백업(Online Backup), 오프라인 백업(Offline Backup)을 한다.

(16) 모니터링 수집

모니터링을 통해 불법적으로 접근을 시도하거나, 침입하는 경로를 파악한다.

(17) 수집 대상 파일

수집한 대상 파일의 변조를 막고 무결성을 획득하기 위해 SHA1/MD5와 같은 해시값(hash)를 기록한다.

디지털 포렌식에서 해시값(hash)은 디지털 지문(Digital Fingerprint)로 사용하므로 무결성을 증명하는데 사용한다.

(18) 체크리스트 작성

데이타베이스 서버에서 자료를 수집하였을 경우 수집 대상에 대한 기록을 남긴다. 체크리스트는 대상 시스템 정보뿐만 아니라 사건 유형과 수집 대상 파일 목록을 기록한다. 수사관이나 분석관이 작성 하도록 한다.

그림 2는 제안하는 데이타베이스 포렌식 모델로 분석 대상을 첫 번째 단계로 목적별로 분류하고 두 번째 단계로 분석기법을 목적에 맞게 순차적으로 재배열한다. 네 가지 모델은 상호보완적으로 연결되어 있으며 분석 목적에 따라 동일한 분석기법도 순서가 다르게 적용되

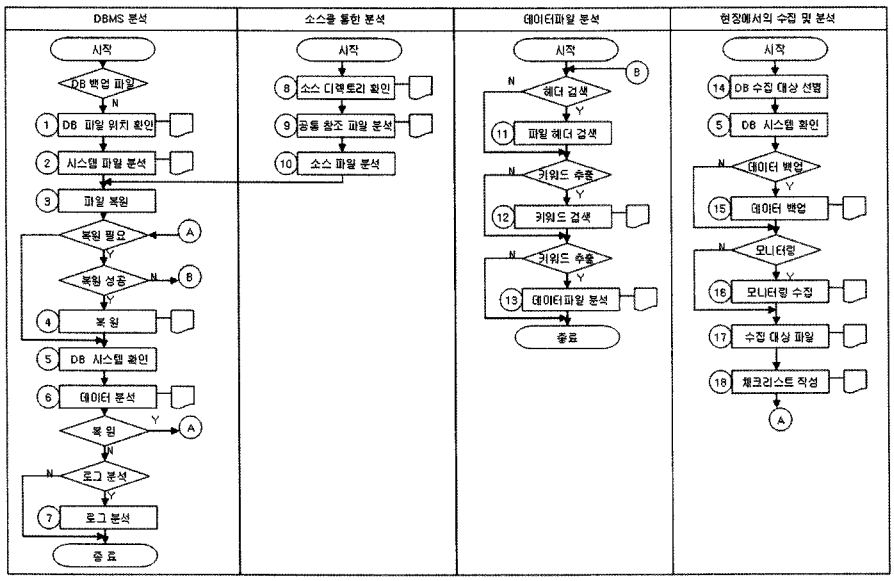

그림 2 제안하는 데이타베이스 포렌식 모델(DFM)

는 것을 확인할 수 있다.

본 논문에서 제안하는 이러한 접근 방법은 다양한 데이타베이스 분석 환경에서 하나의 정형화된 모델과 분석 절차를 동시에 제시할 수 있다.

4. 검증 및 평가

4.1 실험환경

이 실험은 제안하는 데이타베이스 포렌식 모델로 다양한 데이타베이스에 적용한 다음, 이것이 특정 데이타베이스에 종속되지 않고 데이타들의 관계를 분석이 가능함을 보이고자 네 가지 유형으로 나누어 제안 모델을 증명하고자 한다.

각 실험은 데이타베이스 포렌식 모델을 실제 데이타베이스 분석에 적용하여 증명한다.

실험 4.1.1은 DBMS 분석을 여러 가지 데이타베이스에 적용했을 때 벤더에 독립적으로 데이타들의 관계를 분석하여 데이타를 추출하여 검증한다. 실험 4.1.2는 소스분석을 통한 분석으로 프로그램 소스를 통해 데이타베이스의 종류 및 위치정보뿐 아니라 데이타베이스에 저장되지 않은 정보를 확인하고 소스를 분석하여 데이타를 추출 한다. 실험 4.1.3은 데이타파일 분석으로 여러 벤더에서 삭제된 데이타를 데이타파일에서 추출 가능한 것을 증명한다. 실험 4.1.4는 현장에서의 자료를 수집하고 분석하는 절차를 제시한다.

실험을 통해 제안하는 데이타베이스 포렌식 모델은 정형화된 분석기법을 제안함으로써 다양한 분석에서 분석기법이 적절성과 분석기법의 누락 여부를 검증 할 수 있다. 실험 4.1.1, 4.1.2, 4.1.3은 실제 적용하여 분석한 것으로 데이타나 파일의 내용을 일부 모자이크 처리를 한다. 또한 실험들은 표 1에 있는 프로세스를 검증 및 평가에 적용한다.

4.1.1 DBMS 분석 예

DBMS 분석에서는 오라클 9i, MS-SQL 2000과 같이 벤더에 독립적으로 데이타들의 관계를 분석하는 실험을 수행한다. 분석용 데이타베이스 시스템 환경은 다음과 같다.

- 서버 A
  - 운영체제 : Redhat 9.0,
  - 데이타베이스 : 오라클 9i
- 서버 B
  - 운영체제 : Windows 2000 서버
  - 데이타베이스 : MS-SQL 2000

예제 1. 서버 A, B는 분석용 데이타베이스 서버로 간접 분석에 사용한다. 먼저 압수한 오라클의 데이타베이스 원본(suspect)서버에서 운영체제와 데이타베이스의 제품별 특성을 고려하여 DB 파일들의 위치를 파악한 후(PID: 1) 시스템 파일 분석을 통해 접속기록이나 데이타베이스 복구 모델 등을 분석한다(PID: 2). 압수한 오라클의 데이타베이스 원본(suspect)서버에서 DB 파일들을 추출(PID: 3)하고, 서버 A의 데이타베이스 서버로 복원(PID: 4) 하였다. 그리고 데이타간의 관계를 확인하기 위해 프로시저나 트리거 또는 테이블 간 컬럼의 설명을 확인(PID: 5)하였다.

그림 3은 오라클에서 데이타의 관계를 추출하기 위해 직접 개발한 오라클 분석기를 실행한 화면이다. 1라인에서 사용자를 입력하면 사용자가 생성한 테이블의 이름과 행의 수(2, 5, 8, 11, 14, 17라인), 그리고 같이 테이블 설명(12, 15라인)을 확인할 수 있다.

그림 3과 같이 테이블의 설명을 통해 데이타 간의 관계를 분석하였으며, 이는 데이타 분석에(PID: 6) 사용할 수 있다.

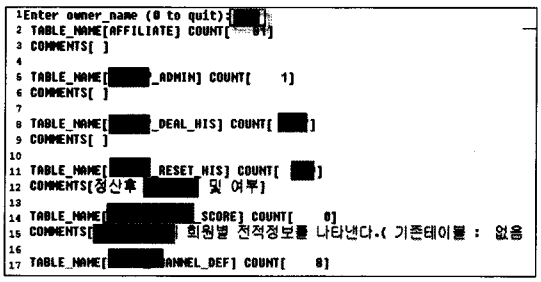

그림 3 오라클의 데이타베이스 시스템 확인

예제 2. MS-SQL의 경우도 운영체제와 데이타베이스의 벤더 특성을 고려하여 DB 파일들의 위치를 파악한 후(PID: 1) 시스템 파일 분석을 통해 접속기록이나 데이타베이스 복구 모델 등을 분석한다(PID: 2). 압수한 MS-SQL의 데이타베이스 원본(suspect)서버에서 DB 파일들을 추출(PID: 3)하고 서버 B의 분석용 데이타베이스 서버에 데이타베이스 복원(PID: 4)하였다. 그리고 데이타간의 관계를 확인하기 위하여 프로시저나 트리거 또는 테이블 간의 컬럼의 설명을 확인(PID: 5)하였다.

그림 4는 게시판, 문서관리, 메일과 관련된 조인 연산 정보가 저장된 프로시저이다. 이를 통해 새로운 SQL이나 프로시저를 생성하여 데이타 분석(PID: 6)을 할 수 있다.

예제 3. 데이타파일이 손상되어 서버에서 복구가 되지 않을 경우 데이타파일 분석으로 분기를 통한 재분석을 해야 한다. 데이타파일 분석에서 파일의 헤더 검색(PID: 11)을 통해 파일을 발견할 수 있고, 키워드 검색(PID: 12)이나 데이타파일 분석(PID: 13)을 통해 데이타를 일부 복구할 수 있는데도 불구하고 지나칠 수 있기 때문에 정형화된 모델이 반드시 필요하다.

```
1 --게시판
2 Select A.UserKey, B.UserName, C.[redacted], D.[redacted],
3 From tblAttach A, tblUser B, [redacted] C, [redacted] D
4 Where ApplKey=1
5 Union All
6 --문서관리
7 Select A.UserKey, B.UserName, C.[redacted] D.[redacted]
8 From tblAttach A, tblUser B,[redacted] C, [redacted] D
9 Where ApplKey=2
10 Union All
11 --메일
12 Select A.UserKey, B.UserName, C.[redacted], D.[redacted]
13 From tblAttach A, tblUser B, tblDept C, tblPost D
14 Where ApplKey=3
```

그림 4 MS-SQL의 데이타베이스 시스템

4.1.1의 예제들의 결과로써 다양한 벤더에 제안하는 분석기법을 적용한 결과 각각의 관계를 분석하고 데이타의 추출이 가능함을 확인할 수 있었다.

4.1.2 소스를 통한 분석 예

본 절은 소스를 통해 데이타베이스의 종류 및 위치뿐 아니라 데이타 관계분석을 위한 정보를 확인한다.

분석용 데이타베이스 시스템 환경은 다음과 같다.

• 서버 C
- 운영체제 : Windows 2000 서버
- 데이타베이스 : MySQL
- 프로그램 언어: PHP

예제 1. 소스를 통한 분석은 시스템과 데이타베이스를 동시에 분석한다. 소스 파일의 디렉토리 위치를 파악(PID: 8) 하고 공통 참조 파일(PID: 9)을 통해 데이타베이스 연결 정보나 데이타베이스 분석에 필요한 정보들을 수집하고 소스 분석을((PID: 10)하여 데이타간의 관계를 분석한다. 그런 다음 데이타베이스 원본(suspect)서버에서 데이타베이스 파일을 추출(PID: 3)하거나 제출 된 데이타베이스 백업파일을 분석용 데이타베이스 서버 C로 복원(PID: 4)한다.

그림 5는 공통 참조 파일 분석을 통해 확장자가 변조된 데이타베이스 정보를 확인한 Include 파일이다. 2라인에서 데이타베이스 파일의 확장자가 asp로 되어 있는 것을 확인할 수 있다. 만약 공통 참조 파일 분석을 하지 않았을 경우 소스 파일로 간주 되어 데이타베이스를 분석을 간과할 수 있다.

```
1 dim Conn,sql,rs,db,connstr
2 db = Server.MapPath("../../[redacted].asp")
3 Set Conn = Server.CreateObject("ADODB.Connection")
4 connstr="Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & db
```

그림 5 확장자가 변조된 공통 참조 파일

예제 2. 예제 1과 같이 먼저 소스 디렉토리를 확인(PID: 8)하고 공통 참조 파일(PID: 9)을 통해 그림 6과 같이 데이타 분석에 필요한 데이타의 속성을 1, 3, 5 라인에서 확인한다.

```
1 array("","주문접수","[redacted]","상품(배송)준비중","상품발송완료"
2
3 array("","상품문의","[redacted]","배송문의","반품/[redacted]문의","
4
5 array("","[redacted]","무통장입금","계좌이체","[redacted]"
```

그림 6 데이타 정보가 저장된 공통 참조파일(PID : 9)

그림 6의 공통 참조 파일 정보(PID: 9)를 확인하고 소스 분석(PID: 10)을 통해 데이타베이스의 관계를 분석한다. 그런 다음 4.1.1의 DBMS 분석으로 분기하여 재분석 한다. 벤더는 다르지만 4.1.1의 예제 1과 같이 압수한 MySQL의 데이타베이스 원본(suspect)서버에서 데이타베이스 파일을 추출(PID: 3)하고 분석용 데이타베이스 서버로 복원(PID: 4)한다.

그림 7은 분석용 데이타베이스 서버에 MySQL을 복원하여 소스에서 분석한 데이타간의 관계를 통해 데이타분석(PID: 6)을 적용한 화면이다.

```
1 SELECT  T.GDS_NAME  AS '상품명'
2        ,T.GDS_STOCK AS '재고량'
3        ,T.TOT_NUM   AS '판매량'
4        ,T.GDS_COST  AS '원가'
5        ,T.GDS_COST * T.TOTAL_NUM AS '원가합계'
6        ,T.GDS_PRI AS '판매가'
7        ,T.GDS_PRI* T.TOT_NUM AS'판매가합계'
8 FROM ( SELECT  SUM(PUR_NUM) AS TOT_NUM, PUR_GDS_ID,GDS_STOCK
9               ,GDS_COST
10              ,GDS_PRI
11              ,GDS_NAME
12        FROM  [redacted]_PUR_TABLE PUR,[redacted]GDS_TABLE GDS
13        WHERE  PUR.REG_DATE >= UNIX_TIMESTAMP('[redacted]')
14        AND    PUR.REG_DATE < UNIX_TIMESTAMP('[redacted]')
15        AND    PUR.APPROVAL_DATE > '10'
16        AND    PUR.PUR_GDS_ID = GDS.UID
17        GROUP BY PUR_GDS_ID
18        ORDER BY TOT_NUM DESC) T
```

그림 7 MySQL의 데이타 분석(PID: 6)

4.1.2의 예제들의 결과는 소스를 통해 데이타베이스의 종류 및 위치와 데이타베이스에 저장되지 않는 데이타 정보까지 확인이 가능함을 확인하였다. 특히 관계(relation) 분석을 할 때 4.1.1에서 제안하는 프로시저나 함수, 테이블간의 컬럼 간 설명(PID: 5)을 활용할 수도 있지만 소스를 통해 더 명확한 데이타간의 관계(relation)를 분석하고 데이타를 추출할 수 있음을 확인하였다.

4.1.3 데이타파일 분석 예

앞서 살펴본 데이타베이스 시스템의 포렌식 분석에서 증명된 것과 같이 삭제된 데이타라도 파일 Slack을 통해 복구할 수 있다. 다만 본 실험에서 복구한 데이타의 수치는 상황에 따라 달라질 수 있지만 제안 모델에는 영향을 주지 않는다. 분석용 데이타베이스 시스템 환경은 다음과 같다.

• 서버 D
- 운영체제 : Windows 2000 서버
- 데이타베이스 : MySQL, 오라클 10g, MS-SQL
- 키워드 검색도구 : Encase 6

데이타베이스 시스템의 포렌식 분석은 제안하는 모델의 하나의 분석(PID: 13)기법에 국한된 문제를 가지고 있다. 본 논문은 이를 더 발전시켜 파일에 저장되어 있는 데이타를 복구하는 방법으로 파일의 헤더 검색 기법(PID: 11)나 키워드 검색(PID: 12)기법을 제안한다.

데이타파일 분석은 데이타베이스 서버를 통해 복원할 필요가 없지만 4.1.3에서는 SQL 질의를 사용하기 위해 데이타베이스 서버를 설치하여 결과를 비교한다. 4.1.3에서는 정상적인 DBMS를 통한 SQL의 결과와 데이타파일에 대한 키워드 결과를 비교함으로써 삭제된 데이타에 대한 복구 가능성을 제시한다. 그리고 SQL 질의를 사용하기 위해 정상적인 데이타파일을 사용하였으나, 손상된 데이타파일에서도 동일한 방법으로 키워드 검색을 사용함으로써 비정상적인 데이타를 복구할 수 있다.

예제 1. 우선 세 가지 데이타베이스 서버에 각 30명의 동일한 주민등록번호를 입력한다. 그리고 SQL을 통해 데이타베이스별로 Select 연산을 한다. 마지막으로 Encase를 통해 세 가지 데이타파일을 동일한 키워드로 검색(PID: 12)한다. SQL 검색과 중복을 제거한 데이타파일을 검색한 결과는 표 2와 같이 동일한 것을 확인할 수 있다.

표 3은 30건의 데이타 중 9건의 데이타를 삭제하고 난후 데이타베이스 서버를 중지하고 데이타파일만 키워드로 검색한 결과이다.

데이타파일 검색 할 경우 삭제된 데이타도 모두 검색되었다. 다만, MS-SQL의 경우 1건의 처음 입력한 데이타가 변경되는 것을 확인할 수 있다.

그 밖에도 키워드 검색은 데이타파일뿐만 아니라 로그파일, 백업 파일을 분석할 때 데이타파일의 전체 구조를 분석하는 것이 아니라 데이타의 특성을 수식이나, 명칭으로 검색하는 기법으로 분석하는데 걸리는 시간을 줄일 수 있다.

4.1.3의 데이타파일 분석결과 역시 데이타베이스의 벤더에 종속되지 않고 데이타를 추출할 수 있으며, 손상된 데이타파일에서도 데이타를 추출할 수 있다. 다만 데이타간의 관계를 통한 추출은 불가능하기 때문에 주로 손상된 데이타파일, 로그파일을 분석하거나 삭제된 데이타를 복원하는 경우 사용한다.

표 2 데이타들을 검색한 결과

데이타베이스	DBMS	Keyword 검색	
	SELECT	검색Hit수	UNIQUE
MySQL 6	30	60	30
오라클 10g	30	30	30
MS-SQL 2000	30	30	30

표 3 데이타들을 삭제한 후 검색한 결과

데이타베이스	DBMS		Keyword 검색	
	DELETE	SELECT	검색Hit수	UNIQUE
MySQL 6	9	21	90	30
오라클 10g	9	21	30	30
MS-SQL 2000	9	21	60	31

4.1.4 현장에서의 데이타 수집 및 분석 예

현장에서 데이타베이스 서버의 데이타를 수집하거나 모니터링하여 데이타를 수집한 경우 무결성을 증명할 수 있다.

현장의 데이타베이스 서버에서 특정 날짜의 데이타만 추출하여야 할 경우는 먼저 여러 대의 데이타베이스 서버 중에서 수집대상 데이타베이스를 선별(PID: 14) 한 후 SQL을 통해 데이타를 백업(PID: 15)한다. 데이타를 백업하고 나서 체크리스트 작성하는데 체크리스트에는 압수하거나 백업하는 데이타베이스 서버의 IP, 인스턴스명, 서버 종류 및 버전, 복구 모델, 인증 방식 등을 기록하며 압수 파일 목록과 압수 파일에 대한 파일 해시값(hash)를 반드시 기록(PID: 18)해야 한다.

예제 1. 표 4는 현장에서 수집 및 분석을 위한 체크리스트이다. 이 때 사용할 수 있는 키워드는 H(Hacking), B(Backup)로 정의한다.

표 4 체크리스트

구분	확인 사항	확인 결과
□H ☑B	서버 IP	10.XXX.XX.XX
□H ☑B	사용 포트	1433
☑B	수집대상 및 분석 데이타베이스명	Sample
□H ☑B	압수 파일 목록 별첨	☑예 □아니요
□H ☑B	압수한 파일 해시값 첨부 유무	☑예 □아니요
□H ☑B	파일 해시 알고리즘	MD5
□H ☑B	수집 일시	2009. 2. X
□H ☑B	분석관	김성혜
□H ☑B	기타 참고 사항	첨부 파일명: Result.txt

4.1.4의 결과는 현장에서 분석대상 서버를 식별하고 필요한 증거를 수집한 뒤 수집된 파일의 목록과 디지털 지문(Digital Fingerprint)인 해시값(hash)을 기록함으로써 무결성을 증명할 수 있다.

압수된 데이타베이스 파일은 그림 2와 같이 A로 분기하고 4.1.1의 DBMS 분석을 통해 데이타베이스를 복원하고 분석한다.

위의 네 가지 실험을 통해 데이타베이스 포렌식 모델의 네 가지 모델을 실제 적용하여 분석할 수 있는지 검증하였다. 본 논문은 다양한 데이타베이스 분석에 분석기법을 통해 분석 시 누락할 수 있는 분석기법을 정형

화하여 데이타베이스 포렌식에서 일반적으로 적용하고자 하였다.

5. 결 론

현재 해킹, 악성코드와 같은 사이버 범죄뿐만 아니라 살인사건, 도난과 같은 일반 사건에도 컴퓨터가 이용된다. 그러므로 범죄 해결을 하는데 컴퓨터에 저장된 전자적 증거가 중요한 역할을 하고 있다.

본 논문은 디지털 포렌식의 여러 분야 중 데이타베이스 포렌식 논문으로 데이타베이스에서 디지털 증거를 추출할 때 사용할 수 있다.

현재까지 데이타베이스가 디지털 포렌식에서 한 분야로 분류되고 있음에도 불구하고, 특정 벤더에 종속적이며 파일시스템에 국한되어 연구가 진행되고 있는 문제점이 있다. 또한 데이타베이스 포렌식의 일반적인 프로세스 모델 및 분석기법에 대한 연구는 진행되지 않고 있기 때문에 제안 논문을 통해 이런 문제를 해결할 수 있다.

본 논문이 제안하는 데이타베이스 포렌식 모델은 첫 번째 단계로 분석 목적에 따라 네 가지 모델로 분류하고 두 번째 단계로 분류된 모델에 따라 18가지 분석기법을 통해 데이타베이스 포렌식의 정형화된 모델과 분석기법을 제안한다.

제안하는 데이타베이스 분석 모델은 다양한 환경에서 일반적으로 사용할 수 있는 모델과 데이타베이스의 증거 수집 또는 분석 시 반드시 필요한 분석기법 정의하였다. 그러므로 제안하는 데이타베이스 분석 모델은 벤더의 물리적인 구조에 종속되지 않고 분석 목적에 따라 데이타베이스를 분석할 수 있다는 장점이 있다.

제안하는 모델을 데이타베이스 포렌식에 실제 적용하여 분석을 수행하였을 때 데이타간의 관계를 분석하고 복구함 할 수 있었을 뿐만 아니라 누락될 수 있는 분석기법을 정형화하여 일관적인 분석 결과를 얻을 수 있다.

본 논문에서는 기존의 특정 벤더에 종속적인 분석과는 다른 접근방식으로 데이타베이스 포렌식 분석 방법에 적용하여 분석의 편리성, 명확성을 향상시켰으며, 향후 프로세스별로 더욱 상세하게 적용시켜 상황별 최적화된 분석 모델을 정의하고자 한다.

또한 데이타베이스 기본적인 시스템 분석의 경우 실험 4.1.1과 같이 자동화가 가능하다. 향후 연구로는 제안 모델의 기본적인 정보를 자동적으로 분석하는 자동화 시스템을 구축하고, 분석결과 및 정보들을 효율적으로 분석할 수 있는 시각화 시스템을 구축할 수 있다. 또한 최적의 분석을 수행하기 위해 다양한 관점으로 분석할 수 있는 방법들에 관한 연구가 필요하다.

참 고 문 헌

[1] George Mohay, Alison Anderson, Byron Collie, Oliver de Vel and Rodney McKemmish, "Computer and Intrusion Forensics," p. 3, Artech House, 2003.

[2] Ed Crowley, "Computer Crime and Forensics," http://isacahouston.org/

[3] Gregory S. Miles, "Computer Forensics: A Critical Process in your incident response plan," BlackHat Europe Briefings, 2001. http://www.blackhat.com/

[4] 이규안, 박대우, 신용태, "포렌식 자료의 무결성 확보를 위한 수사현장의 연계관리 방법 연구", 정보과학회 논문지, 제11권, 제6호, pp.175-184, 2006.

[5] Database Forensics, http://en.wikipedia.org/wiki/Database_Forensics

[6] Encase, http://www.encase.com

[7] FTK, http://www.accessdata.com

[8] David Litchfield, "Oracle Forensics Part 1: Dissecting the Redo Logs," Technical Report, NGSSoftware Insight Security Research(NISR), March, 2007.

[9] David Litchfield, "Oracle Forensics Part 2: Locating Dropped Objects," Technical Report, NGSSoftware Insight Security Research(NISR), March 2007.

[10] David Litchfield, "Oracle Forensics Part 4: Live Response," Technical Report, NGSSoftware Insight Security Research(NISR), April 2007.

[11] David Litchfield, "Oracle Forensics Part 5: Finding Evidence of Data Theft in the Absence of Auditing," Technical Report, NGSSoftware Insight Security Research(NISR), August 2007.

[12] David Litchfield, "Oracle Forensics Part 6: Examining Undo Segments, Flashback and the Oracle Recycle Bin," Technical Report, NGSSoftware Insight Security Research(NISR), August 2007.

[13] Kevvie Fowler, "SQL Server Database Forensics," Balckhat USA briefings and training 2007. http://www.blackhat.com/

[14] Gary L Palmer, "A Road Map for Digital Forensic Research," Technical Report DTR-T0010-01, Digital Forensic Research Workshop(DFRWS), 2001.

[15] Felix Freiling, Heiko Mantel, "Towards Automating Analysis in Computer Forensics," pp.21-56, RWTH Aachen University, 2006.

[16] Venansius Baryamureeba, Florence Tushabe, "The Enhanced Digital Investigation Process Model," Asian Journal of Information Technology, Vol.5, No.7, pp.790-794, 2006.

[17] Nicole Lang Beebe and Jan Guynes Clark, "A hierarchical, objectives-based framework for the digital investigations process," Digital Investigation, Vol.2, No.2, pp.147-167, 2005.

[18] Patrick Stahlberg, Gerome Miklau, Neil Levine, "Threats to privacy in the forensic analysis of database systems," ACM SIGMOD'07, pp.91-102, 2007.

[19] 한국정보통신기술협회(TTA), "Computer Forensics Guideline," TTAS.KO-12.0058, pp.31-33, December 2007.

김 성 혜

2000년 부산외국어대학교 컴퓨터공학과 졸업(학사). 2006년~현재 고려대학교 정보통신 대학원 석사과정. 2000년~2001년 한경 닷컴 개발팀. 2002년~2003년 우리금융정보시스템 카드 IT부. 2004년~2009년 경찰청 사이버테러대응센터 연구원. 관심분야는 디지털 포렌식, 데이타베이스 포렌식, 데이타베이스 보안

김 장 원

2005년 상명대학교 소프트웨어공학과 졸업(학사). 2005년 한국과학기술연구원(KIST) 위촉연구원. 2008년 고려대학교 컴퓨터학과 졸업(석사). 2008년~현재 고려대학교 컴퓨터·전파통신공학과 박사과정. 관심분야는 시맨틱 웹, GIS, 데이타베이스, 메타데이타 등

조 은 애

2003년 고려대학교 컴퓨터학과 학사. 2005년 고려대학교 컴퓨터학과 석사. 2005년~현재 고려대학교 컴퓨터학과 박사과정. 관심분야는 SSL, 접근제어, 권한부여, RBAC, 홈 네트워크, 프라이버시, 유비쿼터스 보안

백 두 권

1974년 고려대학교 수학과(학사). 1977년 고려대학교 대학원 산업공학과(석사). 1983년 Wayne State Univ. 전산학과(석사) 1985년 Wayne State Univ. 전산학과(박사). 1989년~2007년 (사)한국정보과학회(이사/평의원/부회장). 1986년~현재 고려대학교 컴퓨터·전파통신공학과(교수). 1991년~현재 (사)한국시뮬레이션학회(이사/부회장/감사/회장/고문). 1991년~현재 ISO/ IEC JTC1/SC32 전문위원회(위원장). 2001년~현재 (사)도산아카데미(원장) 2002년~2004년 고려대학교 정보통신대학 (초대학장). 2004년~2005년 (사)한국정보처리학회(부회장) 2009년 고려대학교 정보통신대학 학장. 관심분야는 메타데이타, 소프트웨어공학, 데이타공학, 컴포넌트 기반 시스템, 메타데이타 레지스트리, 프로젝트 매니지먼트 등