공학석사 학위논문

디지털 포렌식을 위한 증거 분석 도구의 신뢰성 검증

2010년 2월

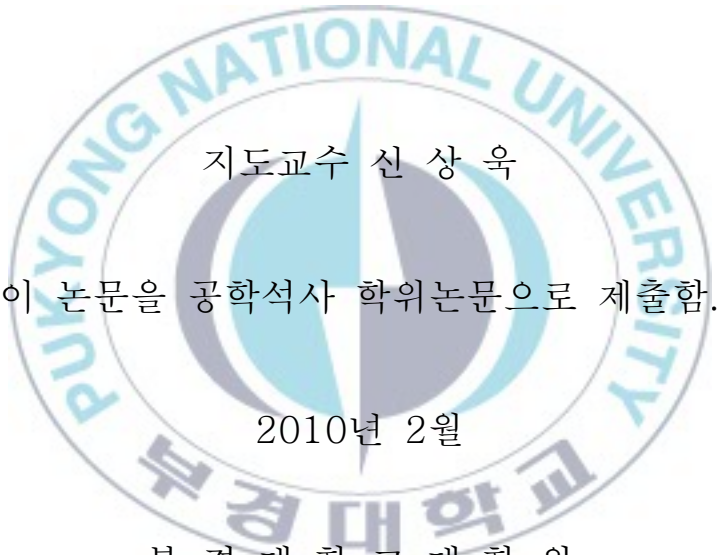
부 경 대 학 교 대 학 원

정보보호학 협동과정

이 태 림

공학석사 학위논문

디지털 포렌식을 위한 증거 분석 도구의 신뢰성 검증



지도교수 신 상 욱

이 논문을 공학석사 학위논문으로 제출함.

2010년 2월

부 경 대 학 교 대 학 원

정보보호학 협동과정

이 태 림

이태림의 공학석사 학위논문을 인준함.

2010년 2월 25일

주심 이학박사 이 경 현 (인)

위원 이학박사 신 원 (인)

위원 이학박사 신 상 욱 (인)

차 례

그림 차례

표 차례

Reliability Verification of Evidence Analysis Tools
for Digital Forensics

Tae Rim Lee

Interdisciplinary Program of Information Security, The Graduate School,
Pukyong National University

Abstract

Because of the wide acceptance of computers and digital devices in our daily lives, it is reasonable to conclude that people will use a computer to commit a crime, store the fruits of their crimes or contraband. So, it is necessary to improve traditional criminal investigation skills and develope new technologies. In this sense, digital forensic is the most promising part to counteract these crimes and digital forensic tool is a key. But in reality, it has limitations in choice of forensic tool for digital evidence acquisition and analysis, because there is no standard for tool's performance assessment or reliability.

This thesis aims to provide test procedure for the verification of reliability of the digital evidence analysis tool. To do this, this thesis defines functional requirements of digital evidence analysis tool through the analysis of various digital evidence analysis tools. And then, verification items for tool's functional requirements are proposed. Also, test procedures and the expected results for each

verification item are proposed and test results on existing forensic tools are presented.

I. 서 론

1. 연구배경

오늘날 우리는 고도로 첨단화된 장비들을 이용하며, 광범위한 네트워크를 통해 전 세계의 컴퓨터들이 서로 연결되어 있는 상황에서 살고 있다. 개인용 컴퓨터는 물론 노트북, PDA, 스마트폰 등은 이미 일상 속에서 널리 사용되어 언제 어디서든 인터넷에 손쉽게 연결할 수 있게 되었으며, 새로운 통신 장비에 대한 시장이 커져감에 따라 우리의 여가 활용법이나 비즈니스 방식은 급격하게 바뀌게 되어 보다 윤택한 삶을 위한 많은 혜택을 누릴 수 있게 됐다[1].

이에 반하여, 모든 것에 장점만이 존재할 수 없듯 첨단 장치 및 관련 IT 기술들이 새로운 범죄의 수단으로 악용됨에 따라 사이버 범죄라는 사회적 문제가 발생하게 되었다. 이는 전 세계적으로도 매년 급증하는 추세로 그 수법 또한 나날이 다양해지고 있으며, 일반적인 범죄에서도 주요 증거 또는 단서가 컴퓨터를 포함한 각종 디지털 장치들에 보관되어 있는 경우가 많아졌다[2]. 이에, 법 집행 기관은 전통적인 수사 방식을 탈피하여, 전자 매체를 이용한 관련 범죄 수사법의 과학적이고 체계적인 개선을 위해 새로운 법적 절차를 마련하고 전문가를 양성하는 등의 지속적인 노력을 하고 있지만 턱없이 부족한 실정이다.

사이버 범죄는 고도의 기술적 범죄라는 특수성으로 인해, 이를 다루기 위해서는 법률 집행 전문가와 IT 전문가 간의 상호 협력이 필요하며, 이를 위한 방법으로 제시할 수 있는 것이 바로 디지털 포렌식

(Digital Forensic) 이다[3].

포렌식이란 범죄수사 또는 증거를 수집하기 위해 과학적이거나 기술적인 기법을 사용하는 것을 의미하며, 이를 위한 도구는 법과 기술 간의 매개체가 될 수 있는 핵심 요소라 할 수 있다[4]. 이는 효율적인 사이버 범죄 수사를 위해 디지털 포렌식 기술 및 도구 개발이 필수적임을 의미하지만, 현재 국내 포렌식 관련 실정은 과도기적 단계로 개념 정립과 수사 절차 확립 및 개선 중에 있으며, 국외의 포렌식 기술과 도구들에 대한 의존성이 강하다[5]. 이러한 상황은 국가적인 낭비일 뿐만 아니라, 수사 도구 선택에 있어서도 매우 제한적일 수밖에 없으므로 국내 환경에 적합하지 않은 요소가 존재할 수 있다. 그리고 기존의 포렌식 도구들에 대한 성능과 신뢰성을 평가할만한 기준 또한 미흡하기 때문에 독자적인 기술 개발과 도구 마련을 위하여 표준을 제시할 수 있는 평가 방안 마련이 시급하다.

본 논문에서는 디지털 포렌식 도구들 중 증거 분석 기능에 초점을 두고 기존에 널리 사용 중인 도구들에 대한 분석을 바탕으로 디지털 증거 분석 도구 요구사항을 제시하며, 이를 바탕으로 대상 도구의 기능 및 동작, 분석 결과물에 대한 신뢰성 검증을 수행할 수 있도록 일반화된 테스트 절차를 제안한다. 또한 실제 상용 포렌식 도구 및 오픈소스 형태의 도구를 대상으로 구체화된 모의 테스트를 설계하고 직접 수행하여, 테스트 모델 제시 및 해당 도구의 장단점을 평가한다.

2. 연구 내용 및 구성

본 논문은 일반적인 디지털 포렌식의 절차로써 증거 평가, 증거 수집, 증거 분석, 문서화 및 보고의 4 단계 중 증거 분석 단계에 초점

을 두어 수집된 데이터로부터 가치 있는 증거 획득을 위해 갖추어야 하는 디지털 증거 분석도구의 기능적 요구사항들을 제시하고, 실제 도구가 해당 요구사항에 얼마나 부합하는지 그 신뢰성을 테스트 할 수 있도록 검증 항목을 정의하며, 각 검증 항목에 대한 테스트 방법 및 절차와 이를 통해 도출되어야 하는 결과를 제시한다. 또한 제시한 테스트 절차를 이용하여 상용 및 오픈 소스 형태의 포렌식 도구를 대상으로 구체화된 모의 테스트를 설계하고 수행하여, 테스트 모델 제시와 활용 방안을 모색한다.

본 논문의 구성은 다음과 같다. 2장에서는 현재 널리 사용되고 있는 상용 도구들과 오픈 소스 증거 분석 도구들에 대한 전반적인 내용과 포렌식 도구 테스트 방법의 하나인 NIST의 CFTT[6] 및 CFReDS[7] 프로젝트에 대한 내용을 다루고, 3장에서는 기존의 분석 도구 기능을 파악하고, 증거 분석 타입들을 분류하여 필수적인 요소들과 세부적인 요소들로 기능 별 디지털 증거 분석도구의 요구 사항을 정립한다. 4장에서는 도구 테스트를 위한 신뢰성 검증 항목을 요구 사항에 맞게 선정하며, 해당 항목 별로 적용 가능한 테스트 절차를 제시한다. 5장에서는 앞서 기술한 테스트 절차를 이용하여 실제 증거 분석 도구를 대상으로 테스트를 수행하고 그 결과를 분석하며, 6장에서는 본 논문의 연구에 대한 결론을 제시하고, 향후 연구 과제에 대하여 언급하며 끝을 맺는다. 마지막 부록에서는 테스트에 사용 가능한 가상의 증거 이미지 생성 방법을 기술한다.

II. 관련 연구

디지털 증거에 대한 과학적인 조사를 주요 내용으로 하는 디지털 포렌식은 탐정과 같은 민간 수사 제도가 발달되어 있고, 적법 절차를 중시하는 영·미 등의 선진국을 중심으로 발전되어 왔으며, 민·형사 소송의 증거 제공에 있어서 핵심적인 역할을 해왔다.

본 장에서는 컴퓨터를 이용한 조사 혹은 컴퓨터와 관련된 조사 과정에서 체계적인 증거확보 절차에 따라 합법적인 증거를 산출해내어 보다 정확한 범죄자 색출 및 범죄 사실 증명을 가능하게 하기 위해 선행 연구되어 왔던 포렌식 관련 연구들을 살펴본다.

1. 디지털 포렌식 증거 획득 및 분석 도구

일반적인 디지털 포렌식 조사 절차는 증거 평가, 수집, 조사, 문서화 및 보고의 4단계로 진행된다. 사이버 범죄 또는 관련 사건이 발생하게 되면 위 단계에 따라 최초 사건 현장에서 범죄에 직접 사용되었거나 연관된 디지털 정보를 포함할 것이라 예상되는 시스템을 평가하고, 이를 대상으로 디지털 데이터를 수집하며, 수집된 데이터를 이용하여 분석한 후 법적 효력을 지니는 형태의 증거로 추출한 다음 선행하였던 모든 조사과정과 최종 증거물을 문서화하여 보고하는 일련의 과정들이 이에 포함된다[8].

이 때 대부분의 과정은 각 단계에서 필요한 작업을 수행하는 디지털 포렌식 도구들을 사용하여 이루어진다. 디지털 포렌식 도구는 특징에 따라 조사 절차상의 각 과정을 독립적으로 수행할 수도 있으며, 전체 과정을 포함하는 하나의 통합 시스템 형태로 존재할 수도 있다.

이에 비추어 볼 때, 성공적인 증거 추출 가능 여부는 조사 과정에서 데이터 획득 및 분석에 사용되는 포렌식 도구의 성능에 따라 좌우된다고 말할 수 있으며, 조사 기관 혹은 조사자는 존재하는 다양한 포렌식 도구들의 기능 및 신뢰성에 대해 명확한 정보를 확보해야 하며, 상황에 적합한 도구를 선택할 수 있는 능력을 갖추어야 한다.

가. 상용 디지털 포렌식 도구

상용화되어 국내외에서 널리 사용되고 있는 대부분의 디지털 포렌식 도구들은 증거 수집부터 분석 및 문서화 과정을 모두 포함하는 통합 도구의 형태를 갖추고 있는 것들이 많다. 대표적인 도구로는 Guidance Software 사의 EnCase[9] 와 AccessData 사의 Forensic Toolkit(FTK)[10] 이 있으며, 그 외에도 ILook Investigator v8[11], ProDiscover[12], SafeBack[13], SMART[14], SDi32[15] 등이 있다.

국내에서 개발된 도구들로는 Final data 사의 Final Forensics[16] 와 A3Security의 A3-AutoWatch[17] 가 있으며, 검찰청에서 개발한 D.E.A.S(Digital Evidence Analysis System For Computer Foresics) 가 있다.

이 밖에도 모바일 기기에 대한 디지털 포렌식 적용을 목적으로 개발된 Paraben[18] 사의 Cell Siezure, PDA Siezure 등의 소프트웨어와 각 종 휴대용 기기들과의 연결을 지원하는 툴박스 형태의 상용 제품들이 제공되고 있다.

나. 오픈 소스 형태의 디지털 포렌식 도구

전반적인 포렌식 조사 절차의 모든 단계를 지원 가능하도록 구성되어진 통합 도구 형태의 상용 디지털 포렌식 도구들과는 달리, 오픈 소스 형태의 디지털 포렌식 도구들은 학문적인 연구의 목적으로 특정 기능이나 운영체제, 장치들을 대상으로 하여 개발되어 오고 있다. 대부분의 도구들은 크게 데이터 획득을 위한 도구와 데이터 분석을 위한 도구로 분류될 수 있으며, 각각 획득과 분석이라는 단일한 목적만을 위해 동작한다. 이는 보다 효율적인 기능 구현과 도구 능력 향상을 위함이며, 지속적인 연구와 업데이트가 이루어지고 있다.

데이터 획득 도구는 기반이 되는 운영체제의 종류에 따라 윈도우즈 기반의 Forensic Acquisition Utilities[19], FTimes[20], liveview[21] 등이 대표적이며, 유닉스 기반의 AFF(Advanced Forensic Format)[22], AIR(Automated Image and Restore)[23], Rdd-2.0.7[24] 등이 있다.

데이터 분석 도구는 주요 분석 대상 및 기법에 따라 세 가지로 분류된다. 그 첫 번째는 미디어 관리 분석 도구로써 The Sleuth Kit(TSK)[25], CDfs[26], CDrecord[27] 등이 있으며, 두 번째는 파일 시스템 기반의 분석도구로 웹 브라우저를 기반으로 한 Autopsy Forensic Browser[28], File System Investigator[29], pyflag[30] 등 이 있다. 끝으로 어플리케이션 기반의 분석 도구로는 Event Log Parser[31], binutils[32], FAUST(File AUdit Security Toolkit)[33] 등 이 있다. 이 외에도 오픈 소스 형태이지만 통합 도구로써 대부분의 기능을 갖추고 있는 네트워크 기반의 포렌식 도구는 F.I.R.E(Forensic and Incident Response Environment)[34] 가 대표적이다.

위와 같이 다양한 오픈 소스 포렌식 도구들이 존재함에도 불구하

고 이들은 여러 가지 제한적인 요소들로 인해 실제 포렌식 조사에서 사용되지 못하고 있다. 그 중 가장 결정적인 이유는 이들 도구를 사용하여 얻게 되는 디지털 증거가 법적인 증거력을 갖추기 위해서 도구에 대한 신뢰성 검증이 필요하지만, 이를 가능하게 하는 제도적인 장치나 표준 등이 마련되어 있지 않다는 것이다. 또한 오픈 소스 프로젝트의 고유 목적에 따라 코드를 업데이트 할 수 있는 개발자의 수가 늘어나야 하지만, 포렌식 프로젝트는 그 특성 상 코드의 열람만을 가능하게 하며 업데이트 권한을 가지는 개발자의 수가 제한적이기 때문에 다수에 의해 지속적인 프로젝트로 진행, 개발되기 어렵다[35].

2. 디지털 포렌식 도구 테스트

가. NIST의 CFTT 및 CFReDS 프로젝트

미국 상무부 산하의 NIST(National Institute of Standards and Technology)에서 진행되고 있는 CFTT(Computer Forensics Tool Testing) 프로젝트[6]는 포렌식 조사에서 사용되는 모든 도구들에 대해 신뢰성 기준을 제공하기 위한 방법을 연구하며, NIJ(National Institute of Justice) 와 법무부 산하 연구 개발 조직 및 OLES(Office of Law Enforcement Std.), ITL(Information Technology Lab.)에 의해 공동으로 진행되고 있다. 또한 FBI, 사이버 범죄 예방 센터 및 U.S. Security Service 를 포함한 여러 관련 기관들에 의해 지원되고 있다.

CFTT 프로젝트는 컴퓨터 포렌식과 연관된 다양한 도구들을 대상으로 하여 그 기능 명세를 밝히고, 필수적인 테스트 요소 및 테스트 범주를 확립하여 기준에 적합한 테스트 절차와 이미지 세트 또는 하드

웨어 개발에 중점을 두고 있다. 이를 통해 디지털 포렌식 도구 제조업자들은 기 개발된 도구의 성능 개선 및 향후 개발될 도구 설계에 있어서 발전적인 방향을 모색할 수 있게 되고, 관련 종사자들은 도구의 능력을 이해하는데 필요한 정보를 제공받을 수 있을 것이다.

현재는 디지털 데이터 획득을 위한 디스크 이미징 장치 대상 테스트 절차 및 요구사항 정립에 이어 소프트웨어 및 하드웨어 쓰기 방지 장치, 삭제된 파일 복구, 모바일 장치, 분석 도구 별 세부 기능 등에 대한 테스트 절차와 이미지들을 개발 중이다. 이에 대한 결과물로 2008년 1월에는 Forensic String Searching Tool Requirements Specification[36] 을 발행하였으며, 이 문서는 디지털 포렌식 수사에 사용되는 문자열 검색 도구를 위한 요구사항들이 정의되어 있다. 요구사항들은 테스트를 통해 검증되어야 하는 도구의 요소들을 언급하고 있으며, 테스트 수행 후 점검해야 하는 상태들을 나열하는 형식이다. 각 사항들은 초기 상태, 테스트 시나리오, 예상 결과들을 상세하게 설명할 수 있는 하나 이상의 테스트 케이스를 통해 검증되어야한다고 밝히고 있다.

또한 2009년 3월에는 Active File Identification & Deleted File Recovery Tool Specification[37] 문서를 통해 삭제된 파일에 대한 인식과 복구를 위해 파일 시스템 메타데이터를 조사하는 디지털 포렌식 도구들의 요구사항을 정의한다. 각 요구사항에 대한 정의 방법과 테스트 관련 내용들은 상위 문서와 유사한 방식을 취한다.

CFReDS(Computer Forensic Reference Data Sets) 프로젝트[7] 역시 CFTT 와 유사한 목적으로 NIST에서 진행되고 있는 프로젝트 중 하나로 이는 실제 범죄 수사에서 획득 가능한 일련의 가상 디지털 증거 이미지들을 제공한다. 이미지의 개발은 주로 CFTT 및

NIJ, OLES 와 공동 작업을 통해 이루어지고 있다.

프로젝트를 통해 개발된 이미지 세트들은 실무 수사 인력 양성 훈련에 사용할 수 있으며, 디지털 증거 획득 및 복구 장비 점검 또는 디지털 증거 분석 도구의 적합성과 품질을 테스트하는데 사용가능하다.

나. Brian Carrier's DFTTI

DFTTI(Digital Forensic Tool Testing Images)[38]는 NIST와 같이 공공 기관에서 이루어지는 포렌식 도구에 대한 테스트 기법 연구와 비영리 단체 또는 학문적 목표를 지닌 개인적 연구들과의 격차를 줄이고자 2003년부터 진행되고 있는 공개 프로젝트이다.

디지털 증거 분석 도구의 핵심 기능에 초점을 맞추어 각 기능 별로 이를 테스트 할 수 있는 가상의 증거 이미지를 공개하고, 웹 사이트 그룹을 통해 위 이미지를 이용한 도구 테스트 결과, 테스트 이미지에 대한 평가 및 개선점, 효율적인 테스트를 위한 의견 등을 수렴하는 형태로 진행되고 있다.

III. 디지털 증거 분석도구 요구사항

본 장에서는 디지털 증거 분석도구가 검증된 디지털 데이터 수집도구를 통해 획득된 데이터로부터 증거를 추출하고 분석함에 있어서 법적인 효력을 갖춘 증거 도출이 가능하게 하기 위한 최소한의 기능적 명세를 제시한다. 이를 위해 먼저 디지털 데이터 증거 분석 작업의 특징을 파악하고 타입을 분류하며, 실제 다양한 형태로 존재하고 있는 증거 분석 도구에 대한 조사를 토대로 일반적인 측면에서와 기능적인 측면에서의 요구사항들을 확립한다. 이는 증거 분석도구의 각 세부 기능에 대한 선행 연구 결과물로 발간되어 II장. 관련연구에서 소개했던 NIST CFTT의 분석도구 기능 별 요구사항 문서 또는 기술 규격서 등을 참조하여, 실제 사용중인 분석도구 모델과 최신 기술 동향, 국내 실정 등을 반영한 것이다.

1. 디지털 증거

디지털 데이터는 변경이나 복제가 용이하다는 특수성으로 인해 적법한 형태의 증거로 인정을 받기 위해서는 이를 다루는 증거 분석도구가 정확하고, 결정적이어야 하며, 검증 가능하게 동작해야 함을 의미한다[39].

하지만 현재 디지털 증거 분석 작업은 도출된 결과에 대한 원인과 방법, 그 특징에 대해 명확히 설명하지 않는 특정 소프트웨어 또는 하드웨어에 의존하고 있으므로, 증거력을 부여하기 위해서는 이를 다루는 조사관 혹은 관련 종사자들이 분석 과정 각 단계의 결과, 도구의 동작 및 원리에 대해 완벽하게 이해하고 있어야 하며, 분석 도구 설계 시 적용된 개발자 고유의 공학적 메커니즘이 공개되어야 한다는 것을 뜻한다.

이는 비현실적인 일이며, 전문가라 할지라도 다양하게 존재하는 디지털 증거 분석도구들에 대해 모두 파악하며, 기능 및 동작 원리에 대한 특징들을 완전히 이해하고 그 신뢰성을 판단하기란 쉽지 않은 일이다.

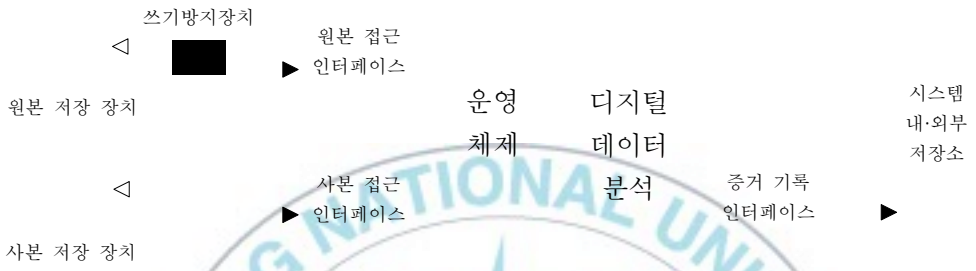
그러므로 디지털 증거 분석도구에 대한 요구사항 확립을 통해 공통의 필수 기능을 정의하고, 정의된 기능에 대한 적합성 테스트를 수행하여 신뢰성이 보장된 도구를 이용하게 함으로써 디지털 데이터에 증거력을 보다 효율적으로 부여할 수 있게 하려는 노력이 필요하다.

또한 증거 분석도구가 증거 추출을 위해 디지털 데이터를 처리함에 있어서도 부가적으로 고려해야 하는 요소들이 있다. 그 중 하나는 대용량 매체 사용이 급증함으로 인해 다루어야 하는 데이터 양 역시 방대해짐에 따라 양의 문제가 발생하는 것인데, 이는 해시 분석 기법 등 알려진 데이터를 분석 대상에서 제외시키거나, 데이터를 큰 이벤트 단위로 그룹화 하여 다루어야 하는 대상 데이터를 축소하는 기술적 개발이 필요하다. 다른 하나는 디지털 데이터 자체가 인간이 직접 이해하기 어려운 가공되지 않은 형태이므로 이를 인식 가능한 형태로 변형함에 있어서 그 복잡도 문제가 발생할 수 있다는 것이다. 이는 디지털 증거 분석 시스템 설계 시 추상화 계층의 개념을 이용하여 어플리케이션에 무관하게 일반적인 비트 형태로 디스크나 네트워크상에 표현되고 저장되는 실제 데이터들을 인식 가능한 타입으로 적절하게 변형시켜줄 수 있게 함으로써 극복하려는 노력이 수반되어야 한다.

2. 디지털 증거 분석도구

디지털 증거 분석도구는 아래 [그림 1]과 같이 구성되며, 직접적인 원본 데이터에 대한 분석을 위해 원본 저장소를 시스템에 부착한 후 해당

인터페이스를 이용하여 증거를 직접 추출하거나, 시스템 내에 사본을 저장한 후 데이터 분석을 수행한다. 원본 데이터의 변경을 방지하기 위해 주로 사본을 통한 증거 분석이 권장 되며, 부득이한 경우 원본에 대해 직접 접근이 필요할 때 쓰기 방지 장치를 이용한다.



[그림 1] 디지털 증거 분석 시스템 구성 예

위 그림에서 사본 저장 장치는 네트워크를 통한 이미지 파일 전송 시 시스템 내 특정 저장소를 이용할 수도 있으며, 운영 체제 및 해당 시스템의 특징에 따라 디지털 증거 분석도구 시스템 구성은 다양하게 나타날 수 있다.

가. 실행 환경

디지털 증거 분석도구의 실행 환경은 시스템 운영체제의 종류, 원본 데이터에 직접 접근 시 원본 저장 장치의 종류와 그에 따다 사용가능한 접근 인터페이스의 종류에 따라 차이가 나타날 수 있으며, 실제 도구 사용 시나 테스트를 위한 구동 시에도 도구 요구 사항들과 도구가 지원 가능한 사항들을 충분히 고려한 환경을 구축할 것을 권장한다.

도구 요구 사항으로는 구동 가능한 운영체제, 소프트웨어가 최상의

성능을 발휘하기 위한 하드웨어 최소 권장 사양 등이 있으며, 도구가 지원 가능한 사항들로는 인식 가능한 파일 시스템의 종류 및 저장 매체 타입, 외부 장치와의 연결을 위해 다양한 장치 종류에 따른 접근 인터페이스 등이 있다.

일반적으로 디지털 증거 분석도구는 보편적으로 사용되고 있는 윈도우, 리눅스 등의 운영체제 상에서 모두 동작하며, 하나 이상의 파일 시스템 및 접근 인터페이스 인식을 지원한다.

다음의 <표 1>는 디지털 증거 분석도구가 실행 가능한 다양한 시스템 환경을 예로 든 것이며, 실제 도구의 실행 환경이 이에 국한되지는 않는다.

<표 1> 시스템 환경 예

분류	운영체제	파일 시스템	저장 매체	접근 인터페이스
종류	MS-DOS Windows 3.1, 95, 98, XP Windows NT series UNIX LINUX MAC OS	FAT12 FAT16, 32 NTFS EXT2, 3 HFS UFS CDFS Palm	플로피 디스크 콤팩트 디스크 하드 디스크 휴대용 저장 매체 기타 저장 매체들	ATA SATA SCSI USB

3. 디지털 증거 분석도구의 주요 기능

디지털 증거 조사에 있어서 분석도구는 기본 원칙에 따라 부득이한 경우를 제외하고 원본 증거에 대한 사본을 대상으로 하여 분석 작업을 수

행하여야 한다. 이 때, 사본이란 수집, 확인 단계에서 신뢰성이 검증된 디지털 증거 수집도구를 이용하여 획득한 데이터들을 의미한다. 증거 조사는 일반적으로 추출과 분석 단계로 이루어지며, 추출은 매체로부터의 데이터 복구, 분석은 복구된 데이터의 해석이나 의미 있는 단위로의 배치 또는 의미 있는 포맷으로 변경하는 과정을 말한다.

가. 추출

(1) 물리적 추출

파일 시스템과 무관하게 물리적 하드 디스크 드라이브 전체를 대상으로 데이터를 식별하고 복구하는 기능을 의미한다. 물리적인 하드웨어를 직접적인 대상으로 하는 키워드 검색이나 파일 카빙 도구들은 운영체제나 파일 시스템에 의해 점유되지 않은 데이터까지 빠짐없이 추출하는데 매우 유용하다. 또한 파티션 구조를 조사함에 있어서 존재하는 파일 시스템을 식별하고 하드 디스크 드라이브 전체의 물리적인 크기가 모두 점유되었는지 등을 판단한다.

(2) 논리적 추출

설치된 운영체제, 파일 시스템, 어플리케이션 등에 기반 하여 파일과 데이터를 식별하고 복구하는 기능을 의미한다. 디렉터리 구조나 파일 속성, 이름, 날짜, 크기, 위치, 시간 정보와 같은 여러 특징을 보여주는 정보들을 추출하는 것이며, 이는 파일 시스템에 종속적이다. 데이터의 단위화는 물론 해시 분석을 이용하여 정보 추출 전 해시 값 비교를 통해 대상 데이터의 조사 범위 축소가 가능하며, 조사 목적과 밀접한 관련이 있는 파일들만을 추려내기 위하여 파일의 이름이나 확장자, 헤더, 내용, 위치

등을 이용한 부분적 추출 역시 가능하다. 삭제된 파일의 복구 및 패스워드로 보호되거나 암호화된 데이터들에 대한 추출 기능이 존재하며, 파일 슬랙 공간 또는 비할당 영역에서의 추출 기능을 제공하는 도구도 존재한다.

나. 분석

추출된 데이터를 해석하는 과정으로 이를 위해 분석도구가 수행하는 기능은 다음과 같다.

(1) 어플리케이션과 파일 분석을 위해 파일의 이름 및 내용, 운영체제의 수와 유형을 확인하고 설치된 어플리케이션에 따라 관련 파일들을 파악한다. 알려지지 않은 파일 유형을 확인하여 보고하며, 시스템 사용자 설정 환경을 분석하여 그에 맞게 파일 메타데이터, 사용자 생성 파일들의 내용을 확인한다.

(2) 파일을 생성하고, 수정, 접근한 주체를 식별하기 위해 파일 소유에 관한 분석 기능을 수행하며, 암호화되거나 패스워드로 보호된 파일에 접근 권한을 얻기 위하여 앞서 획득한 정보들을 바탕으로 분석을 실시한다.

(3) timeframe 분석을 통해 컴퓨터 시스템에서 발생한 이벤트의 시간 및 연관된 사용자 정보들을 분석하며, 파일을 timeframe 에 연동하기 위해서 파일 시스템 메타데이터에 포함된 시간과 날짜 정보를 조사한다.

(4) 시스템 및 어플리케이션의 로그를 분석하며, 에러 및 설치, 접

근, 활용 등과 관련된 로그들이 이에 해당된다. 이 때, 도구의 특징에 따라 개인 컴퓨터의 날짜와 시간 차이의 오차를 고려한 분석 기능을 제공하는 도구도 존재한다.

(5) 숨겨진 데이터를 분석하기 위하여 불일치하는 파일 헤더나 확장자 정보들을 찾아내고,

(6) 압축 및 암호화된 파일에 대한 접근 권한을 획득한다. 널리 알려진 항 포렌식 관련 기술들에 대항할 수 있는 기능을 포함하는 도구도 존재한다.

4. 디지털 증거 분석의 타입 분류

추상화 계층 개념에 기반 하여 디지털 증거 분석도구의 분석 타입을 분류하면 다음과 같다[40]. 일반적으로 디지털 증거 분석도구는 다음의 분류 중에서 적어도 하나 이상의 분석 기능을 지원해야 하며, 지원 가능한 분석 타입을 명시하는 것이 권고된다.

(1) 물리 매체 분석

하드 디스크, 콤팩트 플래시 디스크, 메모리 칩 등 직접적인 물리 매체 추상화 계층에 대한 분석이다. 자체 저장 공간의 레이아웃과 내용들을 IDE 또는 SCSI와 같은 표준적인 인터페이스로 변형하고, 그 경계 계층은 매체의 바이트들이 된다. 위 계층에 대한 분석은 자체 레이아웃 처리, 중복 기록된 이후 삭제된 데이터의 복구 등을 포함한다.

(2) 매체 관리 분석

하드 디스크를 파티션으로 분할하거나 다수의 디스크를 볼륨 단위로 조직화, 다수의 메모리 칩을 메모리 공간으로 통합하는 등 저장 매체를 특정 단위로 조직화하여 수행하는 매체 관리 추상화 계층에 대한 분석이다. 경계 계층은 매체로부터 바이트들의 또 다른 집합이 되며, 모든 유형의 매체에 적용되지 않는다.

(3) 파일 시스템 분석

파일 시스템의 종류와 특징에 따라 내부 디렉터리와 파일들에 대해 조사하고 삭제된 파일을 복구하는 등의 파일 추상화 계층에 대한 분석이다. 파티션의 바이트와 섹터를 디렉터리와 파일로 변형하여 분석하며, 경계 계층은 파일의 내용이 된다.

(4) 어플리케이션 분석

파일 시스템으로부터 반환되는 데이터를 연관되는 어플리케이션에 의해 필요한 자체 포맷의 형태로 변형하는 방법이며, 어플리케이션 추상화 계층에 대한 분석이다. 로그 파일, 환경 설정 파일, 이미지, 문서, 리버스 엔지니어링 실행 파일 조사 등이 이에 포함된다.

다음의 <표 2>는 디지털 증거 분석 분류와 기능 사이의 연관 관계를 나타낸다.

<표 2> 디지털 증거 분석도구의 주요 기능과 분류의 연관 관계

디지털 증거 분석 타입 분류	디지털 증거 분석도구의 기능
물리 매체 분석	HPA 분석
	비트 레벨 삭제 파일 복구
	비트 레벨 스트링 검색
매체 관리 분석	파티션 구조 분석
파일 시스템 분석	파일 시스템 기반의 삭제 파일 복구
	파일 내용, 이름에 대한 스트링 검색
	타임라인 분석
	파일 시스템 메타데이터 분석
	파일 유형(파일 시그네처) 분석
	파일 슬랙 공간 분석
	운영체제 유형, 개수 분석
	파일 내용 분석
어플리케이션 분석	스트링 검색
	웹 히스토리, 캐쉬 파일 분석
	이메일(E-mail) 분석
	로그 파일 분석
	패스워드 검색, 암호 해독
	항 포렌식 관련 분석
	파일 내용 분석

5. 디지털 증거 분석도구의 일반적인 요구사항

디지털 증거가 효력을 가지기 위해서는 디지털 증거를 도출한 분석 도구의 신뢰성이 보장되어야 하며, 해당 도구가 정확하고 객관적인 결과를 일관성 있게 산출한다는 것을 보장하는 능력이 요구된다.

이를 위해, 분석도구는 최대한의 유용성을 제공해야 한다. 이는 디지털 포렌식 조사에 있어서 복잡도 문제와 관련된 것으로 대상이 되는 데이터 타입의 다양성과 수많은 형태의 조사 환경 등을 모두 수용할 수 있어야 함을 의미한다. 조사자가 도구의 규칙과 성능을 벗어나지 않는 한도 내에서 자유롭게 조사를 수행한다면, 도구는 획득한 데이터에 대해 일관

되고 명확한 분석을 실시하며, 그 결과를 조사자가 인식 가능한 형태로 제공해야 한다.

또한 도구는 정확성을 보장해야 하며, 입증 가능해야 한다. 분석 작업을 수행함에 있어서 발생 가능한 오류 문제를 해결하기 위해 도구는 출력 데이터가 정확한 것이며, 적절하게 해석 가능한 오류 한도 내에서 계산한 것임을 보장해야 한다는 것이다. 이와 같은 측면에서 입증 가능함 역시 결과에 대한 정확성을 보조적인 도구 집합을 사용하여 검증할 수 있어야 한다는 것이다. 이를 위해 도구는 각 작업 단계의 입력과 출력에 접근 가능하게 설계되어야 한다.

마지막으로 도구의 분석 결과는 포괄적이며, 결정적이어야 한다. 포괄적이라 함은 데이터 분석을 통해 도구가 도출한 결과가 특정 부분만이 아닌 모든 영역에 걸쳐 누락 없는 분석이 이루어진 것이며, 결정적이라 함은 항상 동일한 입력에 대해 동일한 결과가 나타나야 함을 의미한다.

이 외에도 추가적으로 권고되는 사항들은 도구가 처리하는 데이터와 연관된 것으로 사본 생성과 변경이 용이한 디지털 매체의 특성을 보완하기 위해 항상 읽기-전용으로 하여 다룰 것과 조사자로 하여금 유효한 출력 여부를 구별할 수 있도록 건전도 검사 기능을 지원하는 것들이 있다.

6. 디지털 증거 분석도구의 기능 요구사항

앞서 언급되었던 디지털 증거 분석도구의 주요 기능들, CFTT의 삭제 파일 복구[37] 와 스트링 검색 기능 요구사항[36] 분석을 통해 포렌식 조사 단계에서 증거 데이터 분석을 위해 핵심이 되는 3가지 주요 기능은 다음과 같다.

■ 도구는 분석 대상이 되는 증거 디스크 또는 데이터 이미지 내의 모든 영역에 대해 일체 누락 없는 분석을 수행해야 한다. 범죄자의 수준에 따라 의도적인 데이터 숨김을 위하여 특정 파일 시스템 내에서 슬랙 공간이나 비할당 영역을 이용할 수 있다. 도구는 이러한 점을 고려하여 증거 데이터 검색 시 디스크 전반에 걸친 모든 영역에 대해 해당 기능을 수행할 필요가 있다.

■ 도구는 삭제되거나 변조된 데이터 또는 파일들에 대한 탐지와 복구 기능을 제공해야 한다. 범죄자에 의해 고의로 삭제되거나 변조된 파일들은 대부분 결정적인 증거로 사용될 수 있는 가능성이 높다. 이런 점을 고려할 때, 도구는 다양한 파일 시스템들의 특징을 인지하고 삭제된 파일들을 찾아 복구해 줄 수 있는 능력이 요구된다. 이는 파일 삭제가 실제 디스크에 저장된 데이터의 삭제가 아님을 이용하여 증거 이미지의 파일 시스템에서 삭제된 파일 처리에 대한 특징에 따라 문자열 검색을 통해 탐지하고 복구하도록 구현 가능하다. 또한 변조된 파일의 경우 널리 사용되는 파일들에 대한 특징과 포맷을 사전에 파악하여 대조함으로써 변조 여부를 가려내고, 원본에 가까운 복구를 가능하게 구현할 수 있다.

■ 도구는 특정 문자열 또는 파일들을 탐색 탐색할 수 있어야 한다. 획득된 증거 이미지 내 모든 데이터가 수사에 유효한 것은 아니다. 도구는 증거로써 가치가 있는 데이터만을 효과적으로 찾을 수 있도록 이미지를 분석할 수 있어야 한다. 이는 특정 문자열 검색의 경우 디스크 전반에 걸친 인덱싱 또는 비트 단위 검색을 통해 수행 가능하며, 특정 파일들에 대한 검색의 경우 파일 고유

의 시그너처 분석을 통해 수행 가능하다.

다음 표들은 앞서 3가지로 요약한 디지털 증거 분석도구의 핵심 기능은 물론, 포렌식 도구로써 갖추어야 하는 일반적인 기능들과 보다 원활한 분석 작업을 위해 도구 수준에서 제공 가능한 기타 세부적인 기능들을 제시한 것이며, 이는 디지털 증거 분석도구의 기능 요구사항이라 할 수 있다.

<표 3> 일반적인 기능의 요구사항

구 분	요구사항
GR_01	분석 도구 기능들은 반드시 포렌식적으로 강건한 환경에서 사용되어야 한다.
GR_02	분석 도구 기능들은 입력 데이터의 수정을 최소한으로 유지해야 한다. 만약 입력 데이터의 수정이 발생한다면, 이를 보고해야 한다.
GR_03	분석 도구는 적어도 하나 이상의 파일 시스템을 지원해야 하고, 분석 도구의 기능들은 도구에서 지원되는 파일 시스템과 파일 시스템에서의 메타 데이터를 정확히 인식할 수 있어야 한다.
GR_04	분석 도구는 증거 식별을 위해 증거 매체에 대한 데이터 추출과 분석 기능을 제공해야 한다.
GR_05	분석 도구 기능들은 동일한 환경에서 동일한 입력 데이터가 주어지면 항상 동일한 결과를 출력해야 한다.
GR_06	분석 도구는 수행한 이벤트 로그를 기록할 수 있어야 한다.
GR_07	분석 도구 기능들은 수행 도중 발생한 오류에 대해 보고해야 한다.
GR_08	분석 도구는 분석된 결과에 대한 리포팅이 가능해야 한다.

* GR는 general requirement를 의미한다.

<표 4> 삭제 파일 복구 기능의 필수적인 요구사항

구분	요구사항
DFR_M_01	삭제 파일 복구 기능은 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 파일 시스템에서 복구 기능을 지원해야 한다.

DFR_M_02	삭제 파일 복구 기능은 파일 시스템 객체가 삭제된 이후에 유지되는 메타 데이터에서 복구 가능한 모든 삭제된 파일 시스템 객체들을 식별해야 한다.
DFR_M_03	삭제 파일 복구 기능은 복구된 객체를 구성하는데 있어 발생한 오류를 보고해야 한다.
DFR_M_04	삭제 파일 복구 기능은 잔여 메타 데이터에서 각각의 삭제된 파일 시스템 객체에 대해 복구된 객체를 구성해야 한다.
DFR_M_05	각각의 복구된 객체는 잔여 메타 데이터에서 식별된 모든 할당되지 않은 데이터 블록들을 포함해야 한다.
DFR_M_06	각각의 복구된 객체는 삭제된 블록 풀로부터 데이터 블록들만으로 구성되어야 한다.

* DFR는 deleted file recovery을 의미한다.

* M은 mandatory를 의미한다.

<표 5> 삭제 파일 복구 기능의 선택적인 요구사항

구분	요구사항
DFR_O_01	삭제 파일 복구 기능은 잔여 메타 데이터에서 복구 가능한 복구된 객체의 속성들을 보고해야 한다.
DFR_O_02	삭제 파일 복구 기능이 추정된 콘텐츠를 생성한다면, 각각의 복구된 데이터 블록들은 복구된 객체에 한번 이상 할당되지 말아야 하며, 한번 이상 할당될 경우 이를 보고해야 한다.
DFR_O_03	삭제 파일 복구 기능이 추정된 콘텐츠를 생성한다면, 복구된 객체는 잔여 메타 데이터에서 식별된 원본 파일 시스템 객체로부터의 데이터 블록들로 구성되어야 한다.
DFR_O_04	삭제 파일 복구 기능이 추정된 콘텐츠를 생성한다면, 복구된 객체에서 임의의 데이터 블록들은 잔여 메타 데이터에서 식별된 원본 파일 시스템 객체와 동일한 논리적 순서로 구성되어야 한다.
DFR_O_05	삭제 파일 복구 기능이 추정된 콘텐츠를 생성한다면, 복구된 객체는 원본 파일 시스템 객체와 같은 수의 블록들로 구성되어야 한다.

* O은 optional를 의미한다.

<표 6> 스트링 검색 기능의 필수적인 요구사항

구분	요구사항
SS_M_01	질의에 의해 반환된 응답은 질의에 대한 일치 셋과 같아야 한다.
SS_M_02	한 가지 이상의 문자 인코딩을 사용하여 검색해야 한다.

* SS는 string search을 의미한다.

* M은 mandatory를 의미한다.

<표 7> 스트링 검색 기능의 선택적인 요구사항

구분	요구사항
SS_O_01	스트링 검색 기능이 검색의 전체 영역의 서브셋인 검색 공간의 명시를 허용한다면, 응답에서 모든 검색 일치 결과는 서브셋 검색 공간으로부터 산출된다.
SS_O_02	스트링 검색 기능이 정렬된 서브셋 검색 공간을 검색하고 n의 일치 개수(hit count)가 명시된다면, 응답은 k = min(n, m) 일치를 가진다. 여기서 m은 일치된 집합에서 스트링의 개수이다.
SS_O_03	스트링 검색 기능이 정렬되지 않은 서브셋 검색 공간을 검색하고 n의 일치 개수가 명시된다면, 응답은 k = min(n, m) 일치를 가진다. 여기서 m은 일치된 집합에서 스트링의 개수이다.
SS_O_04	명시된 텍스트 방향으로 질의와 질의응답의 텍스트를 화면에 표시해야 한다.
SS_O_05	영문 키워드에 대한 어간 검색(stemming search)을 수행한다면, Porter stemming 알고리즘[부록 B]의 어간 추출 결과를 사용한 검색의 일치 결과와 유사한 수준이어야 한다.
SS_O_06	동의어 검색(synonym search)을 수행한다면, 질의 스트링의 동의어들이 일치해야 한다.
SS_O_07	퍼지 검색(fuzzy search)을 수행한다면, 질의 스트링의 근접 오철자(close misspelling)들이 일치해야 한다.
SS_O_08	표음식 검색(phonetic search)을 수행한다면, 질의 스트링과 동일하게 발음되는 단어들이 일치해야 한다.
SS_O_09	미리 정의된 질의어를 제공한다면, 반환된 응답은 질의에 대한 일치 셋과 동일해야 한다.
SS_O_10	And, or, not와 같은 논리 연산을 지원할 수 있다.

* O은 optional를 의미한다.

<표 8> 기타 세부 기능 요구사항

구분	요구사항
AF_01	이벤트 로그 기록은 조사관이 수행하는 모든 이벤트를 기록해야 한다. 또한, 수행한 작업의 목록과 순서에 대한 정보를 나열할 수 있어야 한다.
AF_02	분석 결과 리포팅은 증거 이미지에 대한 전체적인 정보이다. 탐지된 증거 데이터에 대한 분석 결과를 리포팅 할 수 있어야 한다.
AF_03	타임라인 분석은 MAC(modified, accessed, change of

	status) 시간 분석을 통해 파일의 생성, 수정, 접근 시간 및 소유권 등의 정보를 추출 가능해야 하며, 이들 정보를 이용해 시간의 순서로 정렬, 나열 가능해야 한다.
AF_04	로그 파일 분석은 적어도 하나 이상의 로그 파일 포맷을 인식할 수 있어야 하며, 원하는 이벤트를 검색, 필터링 할 수 있는 기능을 제공해야 한다.
AF_05	파일 시그네처 분석은 잘 알려진 파일들에 대한 헤더 정보 리스트를 가지고 있어야 하며, 증거 파일의 확장자 변경 등의 위조를 탐지하고 변경 이전의 파일 포맷을 제시할 수 있어야 한다.
AF_06	인터넷 사용 분석은 적어도 하나 이상의 웹 브라우저에 대해 웹 히스토리, 즐겨찾기, 캐시, 쿠키 등의 정보를 추출 가능해야 하며, 원하는 URL, 방문 시간, 사용자 등의 정보를 검색, 정렬 가능해야 한다.
AF_07	이메일 분석은 적어도 하나 이상의 이메일 클라이언트에 대한 데이터 포맷을 인식해야 하며, 송수신된 이메일에 대한 정보를 추출할 수 있어야 한다.
AF_08	시스템/사용자 설정 정보 분석은 운영체제에 대한 정보, 사용자에 대한 정보, 설치된 응용프로그램에 대한 정보들을 분석할 수 있어야 한다.
AF_09	해시 분석은 증거 파일들에 대한 해시 값을 생성, 리스트화가 가능해야 하며, 원하는 해시 값에 대한 검색이 가능해야 한다. 또한 알려진 정상 파일에 대한 해시 값 리스트를 가지고 변경된 파일을 탐지할 수 있어야 한다.

* AF는 additional function을 의미한다.

IV. 증거 분석도구 신뢰성 테스트를 위한 검증 항목 및 절차

본 장에서는 디지털 증거 분석도구와 관련하여 앞서 제시된 요구사항들을 검토한 내용을 토대로 신뢰성 검증을 위해 필수적인 사항들을 테스트하기 위한 검증 항목들을 제시한다. 또한 선정된 검증 항목들을 이용하여 실제 포렌식 분석도구를 대상으로 신뢰성 테스트를 수행할 수 있도록 일반화된 항목별 신뢰성 테스트 절차를 제시한다.

1. 검증 항목 선정

디지털 증거 분석도구의 기능에 초점을 맞추어 작성된 요구사항 표에 따르면 각각의 항목들이 일반적인 포렌식 관점에서의 필수적인 요소들과 증거 조사 절차에서 도구가 수행해야 하는 작업의 특징에 따라 공통적인 요소들을 고려하여 분류된 형태로 제시되고 있다.

이러한 관점에서 도구 기능 검증을 위한 테스트 역시 정확한 증거 분석 작업이 가능하도록 증거물 타입에 대한 올바른 인식 여부와 관련된 일반적인 검증 항목들과 내부 데이터 분석과 관련하여 특정 파일 및 문자열 탐색 기능에 대한 검증 항목, 삭제되거나 변조된 파일에 대한 인식 및 복구 기능에 대한 검증 항목, 조사자의 증거 분석 작업을 보조할 수 있도록 도구가 가질 수 있는 부가적인 기능들에 대한 검증 항목으로 구분하였다.

신뢰성 테스트를 위한 검증 항목을 선정함에 있어서 요구사항들 중 의미상으로 중복되거나 테스트 방법이 중첩되는 사항들은 통합하여

항목화 하였으며, 검증 항목으로 누락된 요구사항들은 테스트 가능성 여부 및 테스트 수준과 관련하여 현실적인 설계가 불가능하거나 무의미한 경우를 배제한 것이다.

선정된 항목들을 이용하여 실제 증거 분석도구를 대상으로 신뢰성 검증 평가를 실시할 경우 모든 검증 항목에 대한 테스트가 이루어져야 하는 것은 아니며, 적용한 모든 테스트를 성공적으로 통과해야만 실무에 적합한 증거 분석도구가 되는 것은 아니다. 이는 테스트 대상이 되는 증거 분석도구가 전반적인 포렌식 조사 절차의 모든 기능을 제공하는 통합 도구 형태일수도 있고, 각각의 독립적인 기능만을 제공하는 도구일수도 있으므로, 그 특징과 주요 기능에 따라 평가 기준이 달라질 수 있으며 지원하지 않는 검증 항목이 존재할 수 있음을 의미한다. 그러므로 실제 성능 평가 시에는 해당 도구의 특징을 정확하게 파악하고, 검증 항목과 부합되는 평가 요소들을 적절히 선택하여 테스트 전략을 수집하며, 적절한 평가 기준을 마련하는 것이 매우 중요하다.

디지털 증거 분석도구의 기능 별 요구사항들에 대한 검증 항목 선정 작업의 내용은 다음과 같다.

가. 일반적인 검증 항목

일반적인 기능에 대한 요구사항을 검증 항목으로 선정하기 위한 검토 내용으로 포렌식 관점에서 필수적인 기초 요소들과 대상 증거물에 대한 정확한 인식 여부 및 도구 실행 환경과 관련된 요소들이 이에 포함된다. GR은 general requirement로 요구 사항 표준의 항목이며, GV는 general verification으로 신뢰성 테스트를 위한 검증 항목에 해당한다.

(1) GR_01. GR_02 => GV_01 : 증거 분석 시에 포렌식적인 관점에서 갖추어야 하는 환경과 분석 대상 데이터에 대한 내용으로 변동과 수정이 발생하지 않음을 검증하는 테스트 항목으로 통합하였다.

(2) GR_01, GR_02 => GV_02 : 증거 분석 작업 환경과 관련 하여 쓰기방지장치가 없는 특수한 환경을 고려하여 추가한 원본 데이터 변경 관련 검증 항목이다.

(3) GR_03~08 => GV_03~08 : 특별한 변동 없이 요구 사항 표의 항목들을 모두 검증 항목으로 선정하였다.

<표 9> 일반적인 검증 항목

구분	검증 항목	관련 요구사항
GV_01	분석 대상 데이터의 변동 및 수정이 발생하지 않음	GR_01, GR_02
GV_02	직접적인 매체 접근 시, 쓰기방지장치가 없는 경우 디지털 데이터 원본이 변경되지 않음	GR_01, GR_02
GV_03	하나 이상의 파일 시스템을 지원하며, 분석 도구가 지원 가능하다고 명시한 파일 시스템과 파일 시스템에서의 메타 데이터를 정확히 인식함	GR_03
GV_04	증거 식별을 위해 증거 매체에 대한 데이터 추출과 분석 기능을 제공함	GR_04
GV_05	동일한 환경에서 동일한 입력 데이터에 대하여 항상 동일한 분석 결과를 출력함	GR_05
GV_06	수행한 이벤트 로그를 정확하게 기록함	GR_06, AF_01
GV_07	수행 도중 발생한 오류에 대해 보고함	GR_07
GV_08	분석된 결과에 대한 정확한 리포팅이 가능함	GR_08

* GV는 general verification을 의미한다.

* GR은 general requirement을 의미한다.

* AF는 additional function을 의미한다.

나. 삭제 파일 복구 관련 기능 검증 항목

포렌식 관점에서 삭제 파일 복구는 먼저 삭제된 파일들을 빠짐없이 정확하게 식별하여 복구하여야 하며, 이를 조사관에게 유용한 형태의 데이터로 제공하기 위해서 수행되어야 하는 많은 동작들이 과연 정확한 것인지 증명되어야 하는 것이 중요하다. 기능 별 요구사항에서는 이와 관련하여 필수적인 기능 요구 사항과 선택적인 기능 요구 사항으로 분류한 뒤 항목들을 제시하고 있으나, 검증 항목 선정에서는 테스트 수준을 고려하여 필수적인 요구사항을 선별하였다. DFR_M 은 deleted file recovery_mandatory 로 요구 사항 표준 항목이고, DFRV 는 deleted file recovery_verification으로 신뢰성 테스트를 위한 검증 항목에 해당한다.

(1) DFR_M_01~03 => DFRV_01~03 : 특별한 변동 없이 요구사항 표준의 항목들을 모두 검증 항목으로 선정하였다.

(2) DFR_M_04~06 => DFRV_04 : 삭제 파일 복구 시 원본 파일과 복구된 파일의 동일성 여부를 테스트하기 위한 항목으로 내용 및 테스트 방법의 유사성을 고려하여 통합하였다.

<표 10> 삭제 파일 복구 관련 기능 검증 항목

구분	검증 항목	관련 요구사항
DFRV_01	도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 파일 시스템에서 모든 파일 타입들에 대한 복구가 가능함	DFR_M_01
DFRV_02	파일 삭제 이후 유지되는 메타데이터에서 복구 가능	DFR_M_02

	한 모든 삭제된 파일들을 식별함	
DFRV_03	삭제 파일 복구 과정에서 파일 재구성과 관련된 오류가 발생한다면, 이를 보고함	DFR_M_03
DFRV_04	복구된 파일들은 원본과 동일함	DFR_M_04, DFR_M_05, DFR_M_06

* DFRV는 deleted file recovery verification를 의미한다.

* DFR은 deleted file recovery를 의미한다.

* M은 mandatory를 의미한다.

다. 파일 및 문자열 탐색 기능 관련 검증 항목

디지털 증거 분석도구의 파일 및 문자열 검색 기능은 사용자와 검색 엔진 사이의 인터페이스를 제공한다. 검색의 전체 영역이라고 할 수 있는 검색 위치는 모든 데이터 영역에 대해 접근 가능해야 하며, 서브셋 영역에 제한될 수도 있다. 검색 결과는 의미 있는 형태로 사용자에게 표현되어야 하는 것이 중요하다. SS_M 은 string searching_mandatory, SS_O 는 string searching_optional 로 요구 사항 표준의 항목이며, SSV 는 string searching verification 으로 신뢰성 테스트를 위한 검증 항목에 해당한다. 파일 탐색 기능의 경우는 기능 별 요구사항 표준에는 별도로 표기되어 있지 않으나, 삭제 파일 복구 및 변조, 숨겨진 데이터 탐색 등의 내용과 유사성을 토대로 문자열 탐색 기능과 접목 시켜 새로운 검증 항목으로 생성하였다. FSV 는 file searching verification 으로 신뢰성 테스트를 위한 검증 항목에 해당한다.

(1) SS_M_01~02 => SSV_01~02 : 특별한 변동 없이 요구 사항 표준의 항목들을 모두 검증 항목으로 선정하였다.

(2) SS_O_01 => SSV_03 : 분석 대상 내의 검색 영역과 관련된 항목으로 모든 영역에 걸친 접근과 분석이 가능해야 한다는 항목이다.

(3) SS_O_01~03 => SSV_04 : 조건이 부여된 검색을 실시할 때, 특히 검색 영역에 있어서 서브셋 영역을 지정하여 검색 시 그 가능 여부와 결과에 대한 정확도 테스트를 하기 위한 항목으로 내용 및 테스트 방법의 유사성을 고려하여 통합하였다.

(4) SS_O_09~10 => SSV_05 : 정규 표현 또는 영문 키워드 검색 등 검색 엔진으로써 추가적으로 제공 가능한 도구의 기능에 대한 항목들을 테스트 방법의 유사성을 고려하여 통합하였다.

<표 11> 문자열 탐색 기능 관련 검증 항목

구분	검증 항목	관련 요구사항
SSV_01	질의에 의해 반환된 응답은 질의에 대한 일치 셋과 동일함	SS_M_01
SSV_02	하나 이상의 문자 인코딩을 사용하여 탐색가능함	SS_M_02
SSV_03	증거 디스크 내 모든 영역에 대한 문자열 탐색이 가능함(비할당 영역, 단편화된 파일 또는 불연속적인 디스크 공간 내 문자열 탐색 등이 가능함을 의미)	SS_O_01
SSV_04	증거 디스크 내 특정 영역에 대한 검색 공간 지정이 가능하다면, 모든 검색 일치 결과는 해당 지정 검색 공간에서만 산출됨	SS_O_01, SS_O_02, SS_O_03
SSV_05	정규 표현식을 이용하거나, 영문 키워드 검색의 경우 대소문자 사용과 무관한 검색이 가능함	SS_O_09, SS_O_10

* SSV는 string searching verification을 의미한다.

* SS는 string searching을 의미한다.

* M은 mandatory를 의미한다.

* O는 optional을 의미한다.

<표 12> 파일 탐색 기능 관련 검증 항목

구분	검증 항목	관련 요구사항
FSV_01	파일 탐색 기능은 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 모든 파일 타입들에 대한 인식 및 탐색이 가능함	GR_03, DFR_M_01, AF_05
FSV_02	증거 디스크 내의 모든 영역에 걸쳐 탐색이 가능함 (단편화된 디스크 섹터, 비할당 영역 또는 슬랙 공간 속의 파일 탐색 등이 수행 가능함을 의미)	DFR_M_05
FSV_03	파일 내부에 포함된 파일 및 삭제된 파일들에 대한 인식 및 탐색이 가능함	GR_04, DFR_M_02

* FSV는 file searching verification을 의미한다.

라. 기타 분석 기능에 대한 검증 항목

디지털 증거 분석도구가 조사자의 원활한 증거 분석 작업을 보조할 수 있도록 부가적인 기능을 제공한다는 가정 하에 해당 도구가 만족해야 하는 요구사항들에 대한 검증 항목이다. AF는 additional function 으로 요구사항 표준의 항목이며, AFV는 additional function verification으로 신뢰성 테스트를 위한 검증 항목에 해당한다.

(1) AF_01 : GR_07 인 오류 보고에 대한 내용과 유사함으로 GV_07로 포함시키고, 검증 항목 선정에서 제외하였다.

(2) AF_03~09 => AFV_02~08 : 특별한 변동 없이 요구 사항 표

준의 항목들을 모두 검증 항목으로 선정하였다.

<표 13> 기타 분석 기능에 대한 검증 항목

구분	검증 항목	관련 요구사항
AFV_01	내부 데이터들에 대한 시간 정보와 관련하여 원본이 작성된 환경에 맞는 분석이 가능함	AF_03
AFV_02	로그 파일 분석은 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 모든 로그 파일 타입들에 대한 인식과 보고가 가능함	AF_04
AFV_03	파일 고유 정보들의 변경에 원래의 파일 타입을 정확하게 인식함	AF_05
AFV_04	인터넷 사용 분석은 다양한 웹 브라우저들에 대해 관련된 기록들을 해당 포맷에 맞게 분석함	AF_06
AFV_05	이메일 분석은 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 모든 이메일 타입들에 대한 인식과 보고가 가능함	AF_07
AFV_06	시스템/사용자 설정 정보 분석을 통해 운영체제, 사용자 및 설치된 응용 프로그램들에 대한 정보의 추출이 가능함	AF_08
AFV_07	해시 분석을 통해 증거 파일들에 대한 해시 값 생성 및 관리가 가능함	AF_09

* AFV는 additional function verification을 의미한다.

2. 검증 항목 별 테스트 방법 및 절차

각 검증 항목들에 대한 테스트는 해당 도구가 동작 가능한 테스트 환경 내에서 독립적으로 수행 가능하며, 일반적인 성능 평가 테스트의 기준에 따라 반복적이고 재현 가능하게 설계되어야 한다. 반복적이라

함은 한 명의 실험자가 동일한 환경과 방법으로 테스트한 결과가 항시 유지됨을 뜻하며, 재현가능이라 함은 한 명 이상의 다른 실험자가 서로 다른 환경에서 테스트를 진행하더라도 동일한 방법을 사용하면 그 결과가 동일하게 나타남을 뜻한다.

테스트 방법 및 절차는 표 형태로 기술한다. 표의 구성은 앞서 선정된 디지털 증거 분석도구의 각 검증 항목들을 나열하고, 해당 항목을 테스트 할 수 있는 방법과 절차를 기술하며, 제시된 절차를 적용한 테스트 수행 시 도출될 것이라 예상되는 결과에 대한 요구사항들을 명시한다.

일반적으로 디지털 증거 분석도구의 기능 검증을 위하여 검증 항목의 테스트 내용을 포함하는 가상의 증거 이미지 파일을 이용하며, 도구가 지원하는 이미지 파일 형식에 맞게 생성한다. 이는 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)을 통해 확인 가능하며, 생성한 이미지 파일에 대한 상세를 테스트 이전에 명시해야 한다. 이미지 파일 생성 방법에 대한 상세한 내용은 부록의 테스트 이미지 생성 방법을 참고하기 바란다.

가. 일반적인 검증 항목에 대한 테스트 절차

■ GV_01 분석 대상 데이터의 변동 및 수정이 발생하지 않음

테스트 방법	가상 증거 이미지 파일을 분석 대상 데이터로 이용하여, 테스트 대상 도구의 분석 작업 전후를 비교한다. 데이터 동일성 검증을 위해서는 여러 가지 증명된 방법들이 사용될 수 있으며, 일반적으로는 전체 데이터에 대한 해시 값을 계산하여 비교한다.
테스트 절차	1. 운영체제 및 도구 설치 등 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 지원하는 형태의 이미지 파일을 이용하여, 해시 값을 구한

	다. 해시 값 비교를 위해서는 암호학적 관점에서 안전성이 검증된 알고리즘을 사용하여야 하며, 대표적인 해시 알고리즘으로는 SHA1, HAS160 등이 있다. 3. 분석 작업 수행 후, 사용한 이미지 파일의 해시 값을 구하여 사전에 구한 값과 비교한다.
예상 결과	테스트 환경 내에서 대상 이미지 파일에 대한 성공적인 분석이 수행되며, 작업 전후의 해시 값이 동일하게 유지된다.

■ GV_02 직접적인 매체 접근 시, 쓰기방지장치가 없는 경우 디지털 데이터 원본이 변경되지 않음

테스트 방법	분석 대상 매체를 직접 연결해서 이용하며, 테스트 대상 도구의 분석 작업 전후를 비교한다. 데이터 동일성 검증을 위해서는 여러 가지 증명된 방법들이 사용될 수 있으며, 일반적으로는 전체 데이터에 대한 해시 값을 계산하여 비교한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축하고, 지원하는 분석 대상 매체에 대해 접근인터페이스를 구성한다. 2. 분석 대상 매체의 해시 값을 구한다. 해시 값 비교를 위해서는 암호학적 관점에서 안전성이 검증된 알고리즘을 사용하여야 하며, 대표적인 해시 알고리즘으로는 SHA1, HAS160 등이 있다. 3. 분석 작업 수행 후, 대상 매체의 해시 값을 구하여 사전에 구한 값과 비교한다.
예상 결과	테스트 환경 내에서 분석 대상 매체에 대한 성공적인 분석이 수행되며, 작업 전후의 해시 값이 동일하게 유지된다.

■ GV_03 하나 이상의 파일 시스템을 지원하며, 분석 도구가 지원 가능하다고 명시한 파일 시스템과 파일 시스템에서의 메타 데이터를 정확히 인식함

테스트 방법	테스트 대상 도구가 지원하는 파일 시스템 형태로 이미지 파일을 생성하고, 분석한 결과 중 메타 데이터 관련 내용을 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축하고, 지원하는 파일 시스템을 확인한다.

	2. 도구가 지원하는 형태의 이미지 파일을 이용하여, 해당 파일을 확인한 파일 시스템 형태로 포맷한다. 3. 이미지 파일 분석 작업을 수행한다. 4. 최종 분석 결과를 확인한다.
예상 결과	지원 가능하다고 명시된 파일 시스템들에 대해 테스트 설정에서 선택한 이미지의 파일 시스템을 정확히 인식하고, 메타 데이터 분석 결과가 사전 설정 내용과 일치한다.

■ GV_04 증거 식별을 위해 증거 매체에 대한 데이터 추출과 분석 기능을 제공함

테스트 방법	특정 데이터를 포함하는 이미지 파일을 이용하여, 도구의 분석 작업을 통해 해당 데이터 추출 가능 여부를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설정을 명시하고, 해당 설정에 맞게 특정 데이터를 포함하는 이미지 파일을 도구가 지원하는 형태로 생성한다. 3. 생성한 이미지 파일을 이용하여, 분석 작업을 수행하고 그 결과를 확인한다.
예상 결과	테스트 설정에 따라 명시된 이미지 파일의 데이터를 분석 결과에서 확인 가능하며, 그 내용이 설정과 일치한다.

■ GV_05 동일한 환경에서 동일한 입력 데이터에 대하여 항상 동일한 분석 결과를 출력함

테스트 방법	테스트 환경을 변화 없이 유지하면서 동일한 입력 데이터에 대해 분석 작업을 반복 수행하여 도구가 도출한 결과들을 비교한다. 동일한 입력 데이터가 유지됨을 보장하기 위해 데이터 동일성 검증을 위한 여러 가지 증명된 방법들이 사용될 수 있으며, 일반적으로는 전체 데이터에 대한 해시 값을 계산하여 비교한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 분석 대상이 되는 동일한 입력 데이터로써 해당 도구가 지원하는 타입으로 하나의 이미지 파일을 생성한다. 3. 생성한 이미지 파일에 대해 최초의 해시 값을 계산한다. 해시 값 비교를 위해서는 암호학적 관점에서 안전성이 검증된 알고리즘을 사용하여야 하며, 대표적인 해시 알고리즘으

	로는 SHA1, HAS160 등이 있다. 4. 생성한 이미지 파일을 대상으로 분석 작업을 수행하여 결과를 도출하고 이를 명시한다. 5. 분석 작업 수행 후 이미지 파일의 해시 값을 다시 계산하고 최초 값과 비교하여 동일한 데이터임을 확인한다. 6. 테스트 환경을 변화 없이 유지하면서 4, 5항의 테스트 절차를 반복하여 복수개의 분석 결과를 명시하고 이를 서로 비교한다.
예상 결과	반복적인 분석 작업을 통해 도출된 결과들의 내용이 모두 동일하다.

■ GV_06 수행한 이벤트 로그를 정확하게 기록함

테스트 방법	테스트 설정을 통해 명시한 증거 분석 작업에 관한 내용들과 실제 도구가 수행하여 기록한 로그의 내용들을 비교한다.
테스트 절차	1. 테스트 대상 도구가 지원 가능한 증거 분석 관련 기능들을 확인한다. 2. 확인된 기능들을 대상으로 도구가 수행할 테스트 분석 작업을 설계하고, 이를 명시한다. 3. 도구가 실행 가능한 환경을 구축하고, 분석 대상이 되는 임의의 이미지 파일을 생성한다. 4. 테스트 설계에 맞게 해당 분석 작업을 정확히 수행한다. 5. 분석 작업이 끝난 후 도구가 생성한 로그를 확인하고, 그 내용을 사전에 명시된 테스트 설계 내용과 비교한다.
예상 결과	테스트 대상 도구를 이용하여 실제 수행한 분석 작업의 내용과 로그에 남겨진 기록들이 정확하게 일치한다.

■ GV_07 수행 도중 발생한 오류에 대해 보고함

테스트 방법	오류가 있는 이미지 파일에 대해 테스트 대상 도구를 이용하여 분석 작업을 수행하고, 이에 대한 오류 보고가 정확하게 되는지 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설정을 통해 오류가 있는 분석 대상 이미지 파일을 이용하고 오류에 대한 내용을 명시한다. 3. 생성한 이미지 파일을 분석하는 과정에서 발생하는 오류에

	대해 도구가 보고하는 결과를 확인한다.
예상 결과	테스트 설정 과정에서 명시된 이미지 파일 오류에 대한 내용들이 분석 과정 중 도구가 보고하는 오류와 일치한다.

■ GV_08 분석된 결과에 대한 정확한 리포팅이 가능함

테스트 방법	테스트 대상 도구가 지원하는 증거 분석 기능을 수행해 보고 그 결과를 정확하게 보고하는지 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 도구가 지원 가능한 분석 기능들을 확인하고, 테스트 설계 시 이를 반영하여 수행할 분석 작업들을 명시한다. 3. 테스트 설계를 바탕으로 분석 대상이 될 이미지 파일을 생성하고, 분석 결과에 나타나야 할 내용들을 명시한다. 4. 분석 작업을 수행한 후 도출된 결과를 2, 3항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	테스트 설계를 통해 수행한 모든 분석 작업에 대한 결과를 빠짐없이 보고하며, 그 내용 또한 테스트 설정과 일치한다.

나. 삭제된 파일 복구 기능 검증 항목에 대한 테스트 절차

■ DFRV_01 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 파일 시스템에서 모든 파일 타입들에 대한 복구가 가능함

테스트 방법	테스트 대상 도구가 지원 가능한 파일 시스템의 삭제된 파일을 포함하고 있는 이미지 파일을 이용하여 복구 기능을 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 도구가 지원 가능한 파일 시스템들을 확인하고, 테스트 설계 시 반영하여 이미지 파일들을 각각의 파일 시스템으로 포맷하고, 이를 명시한다. 3. 복구 대상이 될 일련의 파일들을 준비하고, 포맷된 이미지들을 마운트 시킨 뒤, 준비한 파일들을 복사한 후 삭제한다. 복구 결과에 나타나야 할 내용들을 명시한다. 복사 및 삭제와 관련하여 사용될 파일들의 종류 및 작업 순서 등

	은 테스트 설계 시 자유롭게 설정될 수 있다. 4. 삭제 파일 복구 작업을 수행한 후 도출된 결과를 2, 3항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	테스트 대상 도구가 지원 가능한 파일 시스템 형태의 이미지 파일들에서 테스트 설계에 따라 삭제되었던 파일들이 복구되며 모든 결과가 명시된 사항들과 일치한다.

■ DFRV_02 파일 삭제 이후 유지되는 메타데이터에서 복구 가능한 모든 삭제된 파일들을 식별함

테스트 방법	테스트 대상 도구가 이미지 파일에 대한 분석 기능을 수행하여 삭제 파일 관련 정보들을 해당 파일 시스템 타입에 맞게 보고하는지 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 삭제된 파일들을 포함하고 있는 가상의 증거 이미지 파일을 준비하고, 테스트 전 이미지 내 삭제 파일 및 파일 시스템 관련 정보들을 명시한다. 이 때, 이미지 파일은 테스트 수행자가 직접 설계하여 사용 가능하며, 테스트 설정에 대한 정보들을 명시해야 한다. 3. 준비한 이미지 파일을 이용하여 분석 기능을 수행한다. 4. 도출된 결과 중 삭제 파일 관련 정보들을 2항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	이미지 내 존재하는 삭제된 파일들을 빠짐없이 보고하며, 그 내용 또한 테스트 설정 과정에서 명시한 사항들과 일치한다.

■ DFRV_03 삭제 파일 복구 과정에서 파일 재구성과 관련된 오류가 발생한다면, 이를 보고함

테스트 방법	오류가 있는 대상에 대해 삭제 파일 복구 작업을 수행하고, 이에 대한 오류 보고가 정확하게 되는지 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설정을 통해 오류가 있는 복구 대상을 이용하고 오류에 대한 내용을 명시한다. 오류의 종류는 불량 섹터에 의한 MBR, FAT 손상 및 데이터 영역 손상 등이 있다. 3. 준비한 이미지 파일을 복구하는 과정에서 발생하는 오류에 대해 도구가 보고하는 결과를 확인한다.

예상 결과	테스트 설정 과정에서 명시된 이미지 파일 오류에 대한 내용들이 복구 과정 중 도구가 보고하는 오류와 일치한다.

■ DFRV_04 복구된 파일들은 원본과 동일함

테스트 방법	테스트 전 이미지 파일 내에 존재하는 삭제된 파일들의 해시 값들을 계산하고, 테스트 대상 도구를 이용하여 삭제 파일 복구 작업을 수행한 후 계산된 해시 값들을 비교한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트에 사용할 일련의 파일들을 준비하고, 각각의 파일들에 대한 해시 값들을 구하여 명시한다. 해시 값 비교를 위해서는 암호학적 관점에서 안전성이 검증된 알고리즘을 사용하여야 하며, 해시 알고리즘으로는 SHA1, HAS160 등이 있다. 3. 준비한 파일들을 테스트 설정에 따라 마운트 시킨 이미지 파일에 복사하고, 삭제한다. 이 때, 파일이 복사된 위치와 복사 및 삭제 작업에 관한 상세 절차를 명시한다. 4. 이미지 파일을 테스트 대상 도구를 이용하여 복구하고, 복구된 삭제 파일들의 해시 값을 계산하여 2항의 절차에서 명시한 값들과 비교한다. 5. 최종 복구된 이미지 파일을 분석하여, 3항의 절차를 통해 명시한 내용과 비교한다.
예상 결과	테스트 전 이미지 파일 내 존재하는 사전에 준비한 파일들의 해시 값과 삭제 및 복구 작업 이후 파일들의 해시 값이 모두 일치하며, 파일이 존재하는 위치 및 이미지 파일 구조 모두 동일하다.

다. 파일 및 문자열 탐색 기능 검증 항목에 대한 테스트 절차

■ SSV_01 질의에 의해 반환된 응답은 질의에 대한 일치 셋과 동일함

테스트 방법	특정 문자열을 포함하고 있는 파일이 존재하는 이미지 파일에 대하여 문자열 탐색을 실시하여 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설정을 통해 특정 문자열을 포함한 파일들을 준비

	하고, 이를 탐색 대상 이미지에 복사한다. 준비한 파일들 및 포함한 문자열에 관한 사항들을 명시한다. 3. 문자열 탐색을 수행하기 위한 질의를 만들고, 각 질의에 대해 도구가 도출해야 하는 예상 결과를 명시한다. 4. 테스트 대상 도구로 3항의 절차를 통해 명시한 질의를 이용하여 이미지 파일에 대한 문자열 탐색을 수행하고, 그 결과를 비교한다.
예상 결과	테스트 설정 과정에서 명시된 질의에 대한 도구의 문자열 탐색 결과가 예상 결과와 일치한다.

■ SSV_02 하나 이상의 문자 인코딩을 사용하여 탐색 가능함

테스트 방법	특정 문자열을 각기 다른 문자 인코딩 형태로 저장한 파일들이 존재하는 이미지 파일에 대하여 문자열 탐색을 실시하여 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설정을 통해 특정 문자열을 각기 다른 문자 인코딩을 이용하여 저장한 파일들을 준비하고, 이를 탐색 대상 이미지에 복사한다. 각 파일에서 사용한 문자 인코딩 방법 및 포함한 문자열에 관한 사항들을 명시한다. 3. 테스트 대상 도구를 이용하여 이미지 파일에 대한 문자열 탐색을 수행하고, 탐색 결과를 2항의 절차를 통해 명시한 결과와 비교한다.
예상 결과	문자 인코딩 방법에 상관없이 탐색 하고자 하는 문자열을 정확하게 인식하여 보고한다.

■ SSV_03 증거 디스크 내 모든 영역에 대한 문자열 탐색이 가능함(비할당 영역, 단편화된 파일 또는 불연속적인 디스크 공간 내 문자열 탐색 등이 가능함을 의미)

테스트 방법	분할된 문자열 혹은 파일들이 복사된 이미지 파일을 추가 조작한 뒤, 해당 이미지에 대한 문자열 탐색 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 탐색 대상이 될 문자열을 포함한 일련의 파일들을 준비하고, 적용한 사항들을 명시한다. 테스트에 사용할 문자열 및

	파일 타입 선정은 테스트 설계 시 실험자의 의도에 따라 다양하게 설정 가능하다. 3. 파일들을 이미지 마운트 후 복사 및 삭제, 단편화 시키거나 Hex Editor를 이용하여 특정 이미지 영역에 직접 문자열을 삽입한다. 문자열이 저장된 위치 및 이미지 파일 조작을 위해 수행한 작업들을 명시한다. 이미지 조작을 위한 작업 순서는 테스트 설계 시 자유롭게 설정될 수 있다. 4. 이미지를 이용하여 문자열 탐색을 수행한 후 도출된 결과를 2, 3항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	저장된 디스크 영역에 상관없이 문자열 탐색에 대한 결과가 테스트 설정에 명시된 내용들과 일치한다.

■ SSV_04 증거 디스크 내 특정 영역에 대한 검색 공간 지정이 가능하다면, 모든 검색 일치 결과는 해당 지정 검색 공간에서만 산출됨

테스트 방법	특정 문자열을 포함하는 이미지 파일에 대하여, 검색 공간을 지정한 다음 문자열 탐색을 수행한 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 탐색 대상 문자열을 포함한 이미지 파일을 준비하고, 문자열이 저장된 영역들에 대한 정보를 명시한다. 지정 영역 이외의 탐색 여부를 확인하기 위해 중복된 문자열을 다양하게 배치한다. 3. 테스트 설계에 따라 이미지의 특정 영역에 대해 적용할 문자열 탐색 질의를 만들고, 2항에 명시된 영역 별 저장 문자열 정보를 이용하여 질의에 대한 탐색 결과를 명시한다. 4. 이미지를 이용하여 문자열 탐색을 수행한 후 도출된 결과를 3항의 절차를 통해 명시된 결과와 비교한다.
예상 결과	지정 검색 공간에 따른 질의에 의해, 도구가 탐색한 문자열에 대한 결과가 테스트 설정과 일치한다.

■ SSV_05 정규 표현식을 이용하거나, 영문 키워드 검색의 경우 대소문자 사용과 무관한 검색이 가능함

테스트 방법	정규 표현식 형태와 대소문자의 차이가 존재하는 중복 영문 문자열이 포함된 이미지 파일에 대하여, 문자열 탐색을 수행한 결과를 확인한다.

테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설계에 따라 정규 표현식 적용 및 대소문자 구성이 다른 중복 영문 문자열이 저장된 파일들을 포함한 이미지를 준비하고, 관련 내용을 명시한다. 3. 문자열 탐색을 수행한 후 도출된 결과를 2항의 절차를 통해 명시된 결과와 비교한다.
예상 결과	도구가 탐색한 문자열에 대한 결과가 테스트 설정과 일치한다.

■ FSV_01 파일 탐색 기능은 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 모든 파일 타입들에 대한 인식 및 탐색이 가능함

테스트 방법	테스트 대상 도구가 지원 가능한 파일 시스템의 이미지 파일을 이용하여 파일 탐색 기능을 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 도구가 지원 가능한 파일 시스템들을 확인하고, 테스트 설계 시 반영하여 이미지 파일들을 각각의 파일 시스템으로 포맷하고, 이를 명시한다. 3. 탐색 대상이 될 일련의 파일들을 준비하고, 포맷된 이미지들을 마운트 시킨 후 테스트 설정에 따라 파일들을 복사한다. 파일이 복사된 위치, 준비한 파일들에 대한 정보들을 명시한다. 이미지에 복사될 파일들의 종류 및 작업 순서 등은 테스트 설계 시 자유롭게 설정될 수 있다. 4. 테스트 대상 도구를 이용한 파일 탐색 작업을 수행한 후 도출된 결과를 2, 3항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	파일 탐색 질의에 따른 결과 및 도구가 인식한 파일에 대한 정보들이 테스트 설정에 명시된 내용들과 일치한다.

■ FSV_02 증거 디스크 내의 모든 영역에 걸쳐 탐색이 가능함(단편화된 디스크 섹터, 비할당 영역 또는 슬랙 공간 속의 파일 탐색 등이 수행 가능함을 의미)

테스트 방법	탐색 대상이 될 파일들이 복사된 이미지 파일을 조작하여 특정 파일 삭제 및 단편화를 수행한 후 해당 파일에 대한 탐색

	결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 탐색 대상이 될 일련의 파일들을 준비하고, 포맷한 이미지 파일을 마운트 시킨 후 테스트 설정에 따라 파일들을 복사 및 삭제, 단편화 시킨다. 파일이 복사된 위치 및 이미지 파일 조작을 위해 수행한 작업들을 명시한다. 이미지에 복사될 파일들의 종류 및 작업 순서 등은 테스트 설계 시 자유롭게 설정될 수 있다. 3. 테스트 설정에 따라 각기 다른 영역에 저장된 파일들에 대해 파일 탐색을 수행한 후 도출된 결과를 2항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	파일이 저장된 디스크 영역에 상관없이 파일 탐색 질의에 따른 결과가 테스트 설정에 명시된 내용들과 일치한다.

■ FSV_03 파일 내부에 포함된 파일 및 삭제된 파일들에 대한 인식 및 탐색이 가능함

테스트 방법	탐색 대상이 될 파일들을 변형하여 복사된 이미지 파일을 조작한 뒤, 각 파일에 대한 탐색 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설정에 맞게 탐색 대상이 될 일련의 파일들을 준비한다. 파일 내부에 포함된 파일 형태는 압축된 파일이나 특정 문서 파일에 삽입된 그림 파일 등을 예로 들 수 있다. 테스트에 사용할 파일들의 종류 및 저장 형태는 테스트 설계 시 자유롭게 설정될 수 있으며, 이를 명시해야 한다. 3. 포맷한 이미지 파일을 마운트 시킨 후 테스트 설정에 따라 파일들을 복사 및 삭제한다. 파일이 복사된 위치 및 이미지 파일 조작을 위해 수행한 작업들을 명시한다. 4. 테스트 설정에 따라 각기 다른 형태로 저장된 파일들에 대해 파일 탐색을 수행한 후 도출된 결과를 2, 3항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	파일이 저장되고 조작된 형태에 상관없이 파일 탐색 질의에 따른 결과가 테스트 설정에 명시된 내용들과 일치하며, 원본 파일 타입에 대해 정확히 인식하여 보고한다.

라. 기타 분석 기능 검증 항목에 대한 테스트 절차

■ AFV_01 내부 데이터들에 대한 시간 정보와 관련하여 원본이 작성된 환경에 맞는 분석이 가능함

테스트 방법	FAT 파일 시스템의 시간 처리 특징을 이용한다. 도구가 이미지 내부에 포함된 파일 관련 시간 정보를 정확하게 분석하는지 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 파일 관련 시간 정보를 추출함에 있어서, 표준시와 일광절약시간 적용여부에 따라 도구의 분석 결과 차이가 존재한다는 사실을 이용하기 위하여 여름과 겨울에 생성된 파일을 각각 준비하고, 이들의 생성 시간을 명시한다. 3. 로컬 타임에 의존하는 FAT 파일 시스템 형태로 포맷한 이미지 파일을 마운트 하여, 2항에서 준비한 파일들을 복사한다. 4. 테스트 대상 도구가 지원 가능한 타임 존 설정 관련 기능을 확인하고, 이를 적용하여 이미지 파일 분석을 수행한다. 5. 이미지 내부에 존재하는 파일들의 생성 시간 분석 결과와 2항에서 명시한 내용을 비교한다.
예상 결과	실제 파일이 생성된 시간 정보와 도구가 분석한 파일 생성 시간 결과가 모두 일치한다.

■ AFV_02 로그 파일 분석은 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 모든 로그 파일 타입들에 대한 인식과 보고가 가능함

테스트 방법	테스트 대상 도구가 지원하는 타입의 로그 파일들을 포함하는 이미지 파일에 대한 분석 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 지원 가능한 로그 파일 포맷들을 확인하여 일련의 파일들을 준비하고, 이를 명시한다. 3. 분석 대상으로 사용할 이미지 파일을 마운트 시킨 뒤, 준비한 로그 파일들을 복사한다. 4. 도구의 분석 결과를 2항에서 명시한 내용들과 비교한다.
예상 결과	이미지에 포함된 모든 로그 파일들에 관한 정보와 테스트 대

	상 도구의 분석 결과가 모두 일치한다.

■ AFV_03 파일 고유 정보들의 변경에 대해 원래의 파일 타입을 정확하게 인식함

테스트 방법	파일 고유 정보가 변경된 분석 대상 파일들을 포함하는 이미지 파일에 대한 테스트 대상 도구의 분석 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 테스트 설정에 맞게 파일 시그너처 및 확장자 등 고유 정보가 변경된 일련의 파일들을 준비한다. 시그너처 값의 변경을 위하여 Hex Editor를 사용하며, 적용한 모든 사항들을 명시한다. 테스트에 사용할 파일 종류 및 변경 값들은 테스트 설계 시 자유롭게 설정 가능하다. 3. 분석 대상으로 사용할 이미지 파일을 마운트 시킨 후, 준비한 파일들을 복사한다. 4. 이미지 내 파일들에 대한 분석 작업을 수행한 후 도출된 결과를 2항의 절차를 통해 명시된 사항들과 비교한다.
예상 결과	테스트 설계에 따라 파일이 변형된 형태에 상관없이 원본 파일 타입에 대해 정확히 인식하여 보고한다.

■ AFV_04 인터넷 사용 분석은 다양한 웹 브라우저들에 대해 관련된 기록들을 해당 포맷에 맞게 분석함

테스트 방법	테스트 대상 도구가 지원하는 웹 브라우저들의 사용 관련 정보를 포함하는 이미지 파일에 대한 분석 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 지원 가능한 웹 브라우저들을 확인하여 사용 관련 정보를 저장하는 파일들을 준비하고, 분석 결과에 나타나야 할 내용들을 명시한다. 3. 이미지 파일을 마운트 시킨 뒤, 준비한 파일들을 복사한다. 4. 도구의 분석 결과를 2항에서 명시한 내용들과 비교한다.
예상 결과	이미지에 포함된 모든 웹 브라우저 관련 파일들이 포함하고 있는 정보와 테스트 대상 도구의 분석 결과가 일치한다.

■ AFV_05 이메일 분석은 도구에 의해 제공되는 문서들(사용법, 목적, 동작, 시스템 요구사항 등을 기술하는 자료들의 집합)에 의해 확인된 모든 이메일 타입들에 대한 인식과 보고가 가능함

테스트 방법	테스트 대상 도구가 지원하는 이메일 클라이언트들의 사용 관련 정보를 포함하는 이미지 파일에 대한 분석 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 지원 가능한 이메일 클라이언트들을 확인하여 사용 관련 정보를 저장하는 파일들을 준비하고, 분석 결과에 나타나야 할 내용들을 명시한다. 3. 이미지 파일을 마운트 시킨 뒤, 준비한 파일들을 복사한다. 4. 도구의 분석 결과를 2항에서 명시한 내용들과 비교한다.
예상 결과	이미지에 포함된 모든 이메일 클라이언트 관련 파일들이 포함하고 있는 정보와 테스트 대상 도구의 분석 결과가 일치한다.

■ AFV_06 시스템/사용자 설정 정보 분석을 통해 운영체제, 사용자 및 설치 된 응용 프로그램들에 대한 정보의 추출이 가능함

테스트 방법	독립적인 시스템에 특정 운영체제 및 응용 프로그램 등의 사용자 환경을 구축하고, 이를 디지털 데이터 수집도구로 이미징 하여 테스트 대상 도구를 이용한 분석 결과를 확인한다.
테스트 절차	1. 테스트 대상 도구가 분석 가능한 운영체제를 확인한다. 2. 독립된 시스템에 해당 운영체제를 설치하고, 테스트 설계에 따라 사용자 설정 및 응용 프로그램 설치 등의 사용자 환경을 구축한다. 사용자 환경은 실험자의 의도에 따라 자유롭게 구축될 수 있으며, 관련된 모든 내용들을 명시한다. 3. 디지털 데이터 수집도구를 이용하여 2항에서 구축한 시스템에 대한 이미지 파일을 생성한다. 4. 생성한 이미지 파일에 대한 도구의 분석 결과를 2항에서 명시한 내용들과 비교한다.
예상 결과	테스트 설계에 따라 명시된 시스템의 사용자 환경에 관한 정보들이 도구의 분석 결과와 일치한다.

■ AFV_07 해시 분석을 통해 증거 파일들에 대한 해시 값 생성 및 관리가 가능함

테스트 방법	알려진 해시 값을 가지는 데이터가 포함된 이미지 파일을 테스트 대상 도구로 분석하고 해시 값을 계산하여 비교한다.
테스트 절차	1. 테스트 대상 도구가 실행 가능한 환경을 구축한다. 2. 널리 알려진 해시 값을 가지는 데이터를 포함한 이미지 파일을 준비한다. 이는 해시 리스트를 보유하고 있는 기존의 자료들을 사용 가능하며, 직접 해시 값을 계산한 데이터를 이미지 파일에 복사하여 생성 가능하다. 3. 분석 대상이 될 데이터들의 해시 값들을 명시한다. 4. 테스트 대상 도구로 이미지 파일을 분석하고 내부 데이터들에 대한 해시 값을 계산한다. 5. 계산된 해시 값 결과들을 3항에서 명시한 값들과 비교한다.
예상 결과	도구를 이용하여 계산한 이미지 내부 데이터들에 대한 해시 값들이 사전에 확보한 해시 값들과 모두 일치한다.

V. 디지털 증거 분석도구 신뢰성 테스트

본 장에서는 실제 상용 포렌식 도구 및 오픈 소스 형태의 도구를 선정하여 테스트 대상으로 삼아 앞서 제시된 검증 항목 및 테스트 절차를 활용하여 구체화된 모의 테스트를 설계하고, 직접 수행함으로써 테스트 모델 제시 및 해당 도구의 장단점을 평가한다.

앞서 신뢰성 검증을 위한 항목 선정 시 언급했던 내용과 같이, 본 논문에서 제시한 모든 검증 항목에 대한 테스트를 성공적으로 통과해야만 실무에 적합한 증거 분석도구가 되는 것은 아니다. 그러므로 이하의 테스트 내용은 제안한 테스트 절차에 대한 활용 방법과 하나의 테스트 모델로써 의미를 가지며, 국내 실정 및 다양한 기술적 요소들을 고려해 볼 때 앞으로 보완되어야 할 사항들을 권고함에 의의를 둘 수 있다.

1. 테스트 설계

최초 테스트 대상이 될 디지털 포렌식 도구가 선정되면, 도구가 제공하는 모든 기능들에 대하여 파악한 후 검증 항목 표를 이용하여 테스트 가능한 기능만을 추려낸다.

선출된 항목들을 각각의 테스트 케이스로 분류한 후, 개별 케이스마다 일반적인 테스팅 패러다임에 따라 테스트 설정, 테스트 수행, 결과 분석의 3단계 과정을 거쳐 진행될 수 있도록 설계한다[41].

테스트 설정 단계는 각 케이스의 검증 항목에 해당하는 테스트 절차를 다양한 환경에 맞게 세부적인 내용을 적용 시키는 단계로, 증거 이미지를 생성하고 관련 사항들을 명시하는 작업들이 주를 이룬다.

모든 케이스에 대하여 각각의 테스트 용 증거 이미지 파일이 준비되면 테스트 수행을 통해 도출되는 결과와 발생한 오류 등을 문서화하고, 설정 단계에서 예측했던 테스트 결과와 비교하여 도구의 해당 기능에 대한 성능 평가를 수행한다.

2. 테스트 대상 도구 선정

신뢰성 검증 테스트를 위해 선정한 디지털 포렌식 도구는 다음과 같다.

가. 상용 디지털 포렌식 도구

대부분 사용되고 있는 상용 디지털 포렌식 도구들은 통합 도구로써 이미징 및 기타 분석 관련 기능들을 모두 포함하는 형태이므로, 앞서 제시된 거의 모든 검증 항목들에 걸쳐 실험이 가능하다. 테스트에 사용할 대표적인 디지털 포렌식 도구는 다음과 같다.

(1) EnCase Forensic LE ver 6.0

Guidance Software 사의 제품으로 현재 가장 인지도가 높아 전 세계적으로 사법 및 정부 기관은 물론 상업 회사, 민간에서도 널리 사용하는 법적 유효성을 인정받은 포렌식 솔루션이다.

나. 오픈 소스 디지털 포렌식 도구

상용 디지털 포렌식 도구와 유사한 형태로 통합 도구의 전반적인 기능을 모두 갖춘 도구가 있는 반면, 네트워크 기반이나 문자열 검색

및 파일 복구 등에서 특정한 기능만을 전문적으로 수행하는 도구들이 있다. 사전에 이러한 각 도구들의 특징을 정확히 파악하여 테스트 전략을 수립하는 것이 중요하다.

(1) The Sleuth Kit – 2.09 & AutoPsy – 2.08

오픈 소스 형태의 대표적인 디지털 포렌식 도구이며, 윈도우 및 대부분의 유닉스 계열 OS환경에서 사용가능 하다. 또한 상용 도구에 가까운 통합 도구로써의 기능을 대부분 갖추고 있다. The Sleuth Kit(TSK)은 C 라이브러리 형태이며 The Coroner's Toolkit(TCT) 기반의 커맨드 라인 도구들의 집합이라 할 수 있다. AutoPsy는 TSK를 이용한 분석을 보다 용이하게 하기 위해 TCT, TSK를 연동하여 사용하는 HTML 방식의 그래픽 인터페이스 프로그램이다.

다. 기타 디지털 포렌식 도구

(1) ETRI Forensics

한국전자통신연구원(Electronics and Telecommunication Research Institute, 이하 ETRI)에서는 2007년 3월부터 “정보투명성 보장형 디지털 포렌식 시스템 개발” 과제 수행의 일환으로 고속 포렌식 시스템(High-Speed Forensic System, 이하 HSFS)을 개발하고 있다. HSFS는 고속 데이터 수집 및 검색/분석에 초점을 둔 포렌식 시스템으로 대용량 저장매체를 지원하는 독자적인 이미징 포맷을 가지고 그 속도를 향상시킨다. 또한 삭제 및 유실 데이터의 복구 기능을 제공한다[42].

3. 신뢰성 검증 테스트 및 결과 분석

가. 테스트 대상 기능 선별 및 이미지 파일 생성

신뢰성 검증 대상으로 선정된 EnCase 와 The Sleuth Kit & AutoPsy, ETRI Forensics 는 포렌식 절차에서 필요한 전반적인 기능들을 대부분 갖추고 있는 통합 도구 형태이므로, 검증 항목 표에 기재된 대부분의 항목들에 대한 테스트 수행이 가능하다. 이를 위한 이미지 파일 생성은 테스트 효율성과 도구 특성 등을 고려하여 각 항목별 이미지 파일을 생성하지 않고, 유사성이 존재하거나 동시에 진행 가능한 항목들을 분류 및 통합한 후 구체적인 테스트 설계 내용을 적용하여 생성한다.

(1) 일반적인 검증 항목에 대한 테스트

■ GV_TEST_01

- 관련 검증 항목 : GV_01, 03, 04, 05, 06, 08
- 이미지 파일
  - 이름 : GV_TEST_01.dd (FAT, 10MB)
  - MD5 : B8A9381ADB7A9E296E1AE7D5EE396892

<표 14> GV_TEST_01.dd 내부파일 상세 표

번호	파일	문자열	크기(bytes)	MD5	설명
1	file1.hwp	파일1	14,336	EC0A60216B692906 C9C38577852849BF	한글 파일
2	/directory1/ file2.txt	file2 in directory1	85	B228CE4AC7E05CF2 95426DA91E55DEC3	메모장
3	/directory1/ file3.dat	file3 in directory1	40	D5DF85E7D05AA631 6C8B753D35E3CC3B	

4	/directory2/ taeri.jpg		8,204	087BF89B43907BA8 711E45290158AA95	JPEG 이미지

– 확인 사항 목록

· 분석 작업 수행 후 이미지 파일에 대한 MD5 값 변화 유무

· 파일 시스템 및 이미지 파일 구조, 내부 구성파일 비교

· 문자열 및 파일 탐색 등 수행 후 이벤트 로그 확인

· 분석 리포트 확인, 복수 테스트 수행 후 각 결과의 동일성 비교

■ GV_TEST_02

– 관련 검증 항목 : GV_07

– 쓰기방지장치 없이 연결된 USB 메모리 스틱

· 이름 : MEMORETT USB DISK (FAT32, 967.5MB)

· MD5 : 4D67BB93C9D8DF799E49C0A6AF956C7A

– 오류가 있는 분석 작업 수행을 위하여 외부 장치 데이터에 대한 접근을 시도하고, 분석 도중 외부 장치를 제거하여 고의적인 오류를 발생시킨다.

– 확인 사항 목록

· 분석 작업 수행 중 발생한 도구의 오류 보고를 확인

■ GV_TEST_03

– 관련 검증 항목 : GV_02, 03

– 쓰기방지장치 없이 연결된 USB 메모리 스틱

· 이름 : MEMORETT USB DISK (FAT32, 967.5MB)

· MD5 : 4D67BB93C9D8DF799E49C0A6AF956C7A

– 확인 사항 목록

· 분석 작업 수행 후 USB에 대한 MD5 값 변화 유무

· 파일 시스템 및 내부 구조(디렉터리, 파일) 비교

(2) 삭제 파일 복구 기능 검증 항목에 대한 테스트

■ DFRV_TEST_01

– 관련 검증 항목 : DFRV_01, 02, 04

– 이미지 파일

· 이름 : DFRV_TEST_01.dd (FAT, 10MB)

· MD5 : 1E58DB962FA63EDFF60950766B4DF01F

<표 15> DFRV_TEST_01.dd 내부파일 상세 표

번호	파일	크기(bytes)	MD5	설명
1	single_cluster.dat	6	DD5C07036F2975FF4BCE568B6511D3BC	단일 클러스터
2	multi_cluster.dat	75,303	A89E077ADF8C7B4BC4CA7CEE4608905F	다수 클러스터
3	single_cluster.txt	6	DD5C07036F2975FF4BCE568B6511D3BC	txt 텍스트
4	/dir1			삭제된 디렉터리
5	/dir1/in_deleted_dir1.txt	11,284	6D4C089E8492EEAA3B53AA31AA6C61CF	삭제된 디렉터리 내 파일
6	/dir1/dir2			삭제된 디렉터리
7	/dir1/dir2/한글파일_in_deleted_dir2.hwp	13,824	2F7C17F46EC78B3D20ADDE57C9453C4C	삭제된 디렉터리 내 파일

– 확인 사항 목록

· 해당 이미지에 대한 파일 시스템과 메타 데이터 정보 비교

· 식별된 삭제 파일 및 디렉터리 목록과 구조 비교

· 원본과 복구된 파일들의 MD5 값 대조를 통한 동일성 비교

■ DFRV_TEST_02

– 관련 검증 항목 : DFRV_03

– 이미지 파일

· 이름 : DFRV_TEST_02.dd (FAT, 5.9MB)

· MD5 : 4AEB06ECD361777242AB78735D51ACE6

<표 16> DFRV_TEST_02.dd 내부파일 상세 표

번호	파일	크기(bytes)	MD5	비고
1	sing.dat	780	59B20779F69FF9F0AC5FCD2C38835A79	단일 클러스터
2	mult1.dat	3,081	FFD27BD782BDCE67750B6B9EE069D2EF	다수 클러스터
3	frag1.dat	1,584	7A3BC5B763BEF201202108F4BA128149	단편화된 파일
4	frag2.dat	3,873	0E80AB84EF0087E60DFC67B88A1CF13E	단편화된 파일
5	/dir1			삭제된 디렉터리
6	/dir1/mult2.dat	1,715	59CF0E9CD107BC1E75AFB7374F6E05BB	삭제된 디렉터리 내 파일
7	/dir1/dir2/			삭제된 디렉터리
8	/dir1/dir2/frag3.dat	2,027	21121699487F3FBBDB9A4B3391B6D3E0	단편화된 파일

– 확인 사항 목록

· 원본과 복구된 파일들의 MD5 값 대조를 통한 동일성 비교

· 비정상적으로 복구된 파일들에 대한 도구의 보고 여부

(3) 파일 및 문자열 탐색 기능 검증 항목에 대한 테스트

■ FSV_TEST_01

– 관련 검증 항목 : FSV_01, 02, 03

– 이미지 파일

· 이름 : FSV_TEST_01.dd (FAT, 10MB)

· MD5 : E4B7A7E7E474981DAD87098EF69584E8

<표 17> FSV_TEST_01.dd 내부파일 상세 표

번호	파일	크기(bytes)	MD5	설명
1	MSOffice.docx	11,475	49C1A7E40CA3CFBF8D8F7239A4313F9C	DOC
2	PPT.pptx	527,240	9B8CF3F55EE4CFFB9BF7B57D977ACA40	PPT
3	TXT.txt	6	515284BF34265A3E4647A5200D6FC065	TXT
4	taeri_jpeg.jpg	8,204	087BF89B43907BA8711E45290158AA95	JPEG
5	taeri_png.png	40,917	74F2EC576742891DC8824193CCA5C132	삭제된 PNG
6	netcfv35.messages.ko.wm.cab	270,954	F59BD607FFF748B334BD26833C2EA52C	CAB
7	clock.avi	82,944	BB516947768FBB05B41A2487F200716E	AVI
8	LoopyMusic.wav	940,794	E2FA75ADE398C9A44815B11CC141105C	WAV
9	filedatech.zip	20,710	BC65324FBAC4A8974665996E703391B9	ZIP
10	FileDate.exe	49,152	7E80E6205C1EDF29CCCD37297049755C	압축된 EXE
11	readme.txt	3,489	E0C646E43430725FCDE6E68DA667FCEE	압축된 TXT
12	/dir1			디렉터리
13	taeri_bmp.bmp	27,538	691A35F3CA24FFFF10EEF0283B0A0D79	디렉터리 내 파일
14	taeri_gif.gif	7,076	09236F0B6589B0BA6041BEB03FD5ACCB	디렉터리 내 삭제

– 확인 사항 목록

· 해당 이미지에 대한 파일 시스템과 메타 데이터 정보 비교

· 인식된 파일 종류 및 관련 정보, 디렉터리 구조 비교

■ SSV_TEST_01

– 관련 검증 항목 : SSV_01, 02, 03, 04, 05

– 이미지 파일

· 이름 : SSV_TEST_01.dd (FAT, 15MB)

· MD5 : CC247D7E2D2B9BD9E0AEBE6199E12BFE

<표 18> SSV_TEST_01.dd 내부파일 상세 표

번호	파일	문자열	문자열 위치
1	file1.dat	first	파일 내
2	file2.dat	SECOND	파일 내
3		SECOND	디렉터리 엔트리 내 파일명
4	file1.dat & file2.dat	1cross1	할당된 두 파일 간 교차
5	file3.dat	2cross2	한 파일 내 섹터 간 교차
6		3cross3	비할당 영역
7	file2.dat & file2 slack	1slack1	파일과 슬랙 영역 간 교차
8	file3 slack & file4.dat	2slack2	슬랙 영역과 파일 간 교차
9	file4 slack	3slack3	슬랙 영역
10	file4.dat	1fragment1	단편화된 섹터 간 교차
11	file6.dat	2fragment sentence2	단편화된 섹터 간 교차
12	file5.dat	deleted	삭제된 파일 내
13	file7.dat	a?bWc*d$e#f[g^	파일 내 정규표현식
14	ANSIHangul.dat	한글	파일 내
15	UTF16LEHangul.dat	한글	파일 내
16	UTF16BEHangul.dat	한글	파일 내
17	UTF8Hangul.dat	한글	파일 내

– 확인 사항 목록

· 표기된 문자열에 대해 누락 없는 탐색 가능 여부

· 탐색 질의 및 탐색 조건 지정에 따른 결과 확인(탐색 공간 지정, 정규 표현식, 대소문자 지정 영문 검색 등)

(4) 기타 분석 기능 검증 항목에 대한 테스트

■ AFV_TEST_01

– 관련 검증 항목 : AFV_01

– 이미지 파일

· 이름 : AFV_TEST_01.dd (FAT, 1.4MB)

· MD5 : 9FB582F3361BA0BC5A3B0F7C17A082CB

<표 19> AFV_TEST_01.dd 내부파일 상세 표

번호	파일	파일 생성 시간 정보
1	winter.txt	1월 1일, 2004년, 오후 2:00
2	summer.txt	6월 1일, 2004년, 오후 3:00

– 확인 사항 목록

· 분석 결과의 파일 생성 시간 정보 비교

■ AFV_TEST_02

– 관련 검증 항목 : AFV_02, 03, 04, 05, 06, 07

– 이미지 파일

· 이름 : AFV_TEST_02.dd (NTFS, 4.0GB)

· MD5 : 0F3206FB0214DC75CB5A04FA5C325410

· 분석 대상이 될 가상 시스템을 구축하기 위해서 VMware workstation 을 이용하여 Windows XP 를 설치하고, 인터넷 및 이메일 사용 기록과 아래 표에 기재된 파일들을 저장한다.

<표 20> AFV_TEST_02.dd 내부파일 상세 표

번호	파일	MD5	설명
1	file1.jpg	087BF89B43907BA8 711E45290158AA95	정상적인 JPEG 파일
2	file2.dat	087BF89B43907BA8 711E45290158AA95	확장자를 변경한 JPEG 파일
3	file3.jpg	59188EAD8BB1FAEE 7B455E449D6697B0	확장자만 jpg인 랜덤 파일
4	file4.jpg	302A7C427DA0D46E A952F5E6B2133171	JPEG 헤더 시그너처를 삽입한 랜덤 파일
5	file5.abc	0261963242290FBE6 E4DA035AC2E9843	JPEG 헤더 시그너처를 여러 곳에 삽입한 랜덤 파일
6	file6.jpg	087BF89B43907BA8 711E45290158AA95	삭제된 JPEG 파일
7	file7.dat	087BF89B43907BA8 711E45290158AA95	확장자 변경 후 삭제된 JPEG 파일
8	file8.dat	087BF89B43907BA8 711E45290158AA95	확장자를 변경한 JPEG 파일
9	file8.zip	3496F442D30DD2D8 E57BCE919560172D	file8.dat를 포함한 ZIP 파일
10	file8.abc	3496F442D30DD2D8 E57BCE919560172D	file8.dat를 포함하며, 확장자를 변경한 ZIP 파일

· 데이터 수집 도구를 이용하여 전체 디스크를 이미징한다.

– 확인 사항 목록

· 해당 이미지의 운영 체제, 시스템/ 사용자 정보, 설치된 프로그램, 파일 시스템 관련 정보를 확인

· 고유 정보 변경 파일에 대한 원래의 파일 타입 인식 여부

· 인터넷 및 이메일 사용 기록, 운영체제의 로그 파일 분석 여부

· 분석 결과에 나타난 내부 파일들과 원본 파일의 MD5 값 비교

나. 증거 분석도구 테스트 결과

테스트는 대상 증거 분석 도구를 각각의 동작 환경에 적합한 시스템에 설치 후, 해당 테스트 케이스에 맞게 별도로 사건 파일을 만든 다음 생성한 이미지 파일들을 raw image 타입의 증거물로 등록하여 증거 분석을 수행한 결과이다.

(1) EnCase Forensic LE ver 6.0

■ 시스템 정보

Intel(R) Core(TM)2 Quad CPU Q9300 2.50GHz, 3GB RAM
Microsoft Windows XP Professional Service Pack3

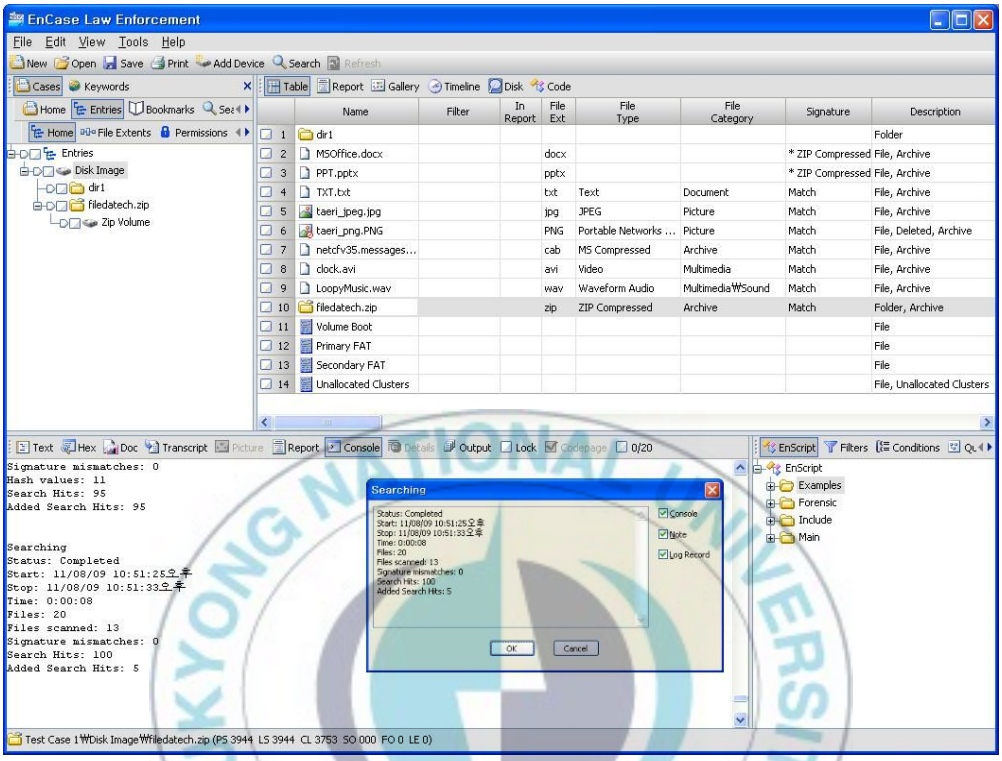
<표 21> EnCase Test 결과

Test Cases	결 과
GV_TEST_01	만족
GV_TEST_02	만족
GV_TEST_03	만족
DFRV_TEST_01	만족
DFRV_TEST_02	불만족
FSV_TEST_01	만족
SSV_TEST_01	불만족
AFV_TEST_01	만족
AFV_TEST_02	만족

- GV_TEST_01 : 분석 작업 수행 결과를 <표 14>와 비교해 본 결과 이미지 파일 구조, 파일 시스템, 내부 구성 파일에 대한 상세 정보가 모두 일치하였으며, 문자열 및 파일 탐색 등 수행한 모든 작업에 대한 이벤트 로그를 확인 가능하였다. 또한 동일한 분석 작업을 복수 수행한 결과가 모두 일치하였으며, 이미지 파일에 대한 MD5 값 변화가 발생하지 않았다.

- GV_TEST_02 : 분석 작업 수행 시 고의로 제거된 외부 장치에 대한 오류 메시지를 정확하게 보고하였다.
- GV_TEST_03 : USB 디스크에 대한 분석 결과가 모두 일치하였으며. 분석 전 후의 MD5 값의 변화가 없으므로 쓰기방지장치 없이 연결되었음에도 데이터 변경이 이루어지지 않았음을 확인하였다.
- DFRV_TEST_01 : EnCase 가 식별한 삭제 파일 및 디렉터리 목록이 <표 15>에 기재된 내용과 모두 일치하였으며, 데이터 복구 후 MD5 값 비교를 통해 원본과 동일함을 확인하였다.
- DFRV_TEST_02 : <표 16>에 기재된 삭제된 파일들을 모두 식별하기 하였으나, 복구 후 단편화 된 파일 frag1.dat, frag2.dat, frag3.dat 의 MD5 값이 원본과 일치하지 않았다.
- FSV_TEST_01 : 분석 결과에 나타난 이미지 파일에 대한 파일 시스템 및 메타 데이터, 디렉터리 구조, 내부 구성파일 종류와 관련 정보들이 <표 17>과 모두 일치하였다.
- SSV_TEST_01 : 문자열 탐색 질의를 통해 <표 18>에 나타난 대부분의 문자열들에 탐색이 가능하였으나, 한글 관련 검색에 있어서 ANSIHangul.dat 파일 내의 사항이 누락되었다. 키워드 검색 옵션 내에 ANSI Latin-1 이란 항목은 있었으나 테스트에 포함된 인코딩 타입과는 차이점이 존재하여 나타난 결과라 생각된다.
- AV_TEST_01 : 분석 결과에 나타난 두 파일에 대한 파일 생성 시간 정보가 <표 19>에 기재된 내용과 일치하였다.
- AV_TEST_02 : 해당 이미지 파일의 운영 체제, 시스템/ 사용자 정보, 설치된 프로그램들이 모두 테스트 설계와 일치하게 분석되었으며, 고유 정보가 변경된 파일 탐색 결과 또한 <표 20>의 내용과 일치하였다. 이미지 내부의 각 파일에 대한 해시 분석이 가능함을 확

인하였다.



[그림 2] EnCase Test 화면

EnCase 의 경우 대부분의 테스트에서 만족스러운 분석 결과를 보여 주었다. 하지만 DFRV_TEST_02 에서 알 수 있듯이, 단편화된 파일 처리 및 복구 기능에 대한 개선이 필요하며, 파일 브라우징 측면에 있어서 최신 버전의 hwp 파일을 제대로 인식하지 못하고, ANSI 인코딩 타입 한글 문자를 탐색하지 못하였으므로 이에 대한 업데이트가 필요하다.

(2) The Sleuth Kit – 2.09 & AutoPsy – 2.08

■ 시스템 정보

Intel(R) Core(TM)2 Quad CPU Q9300 2.50GHz, 3GB RAM
Microsoft Windows XP Professional Service Pack3
Ubuntu in VMware workstation 6.0

<표 22> TSK & AutoPsy Test 결과

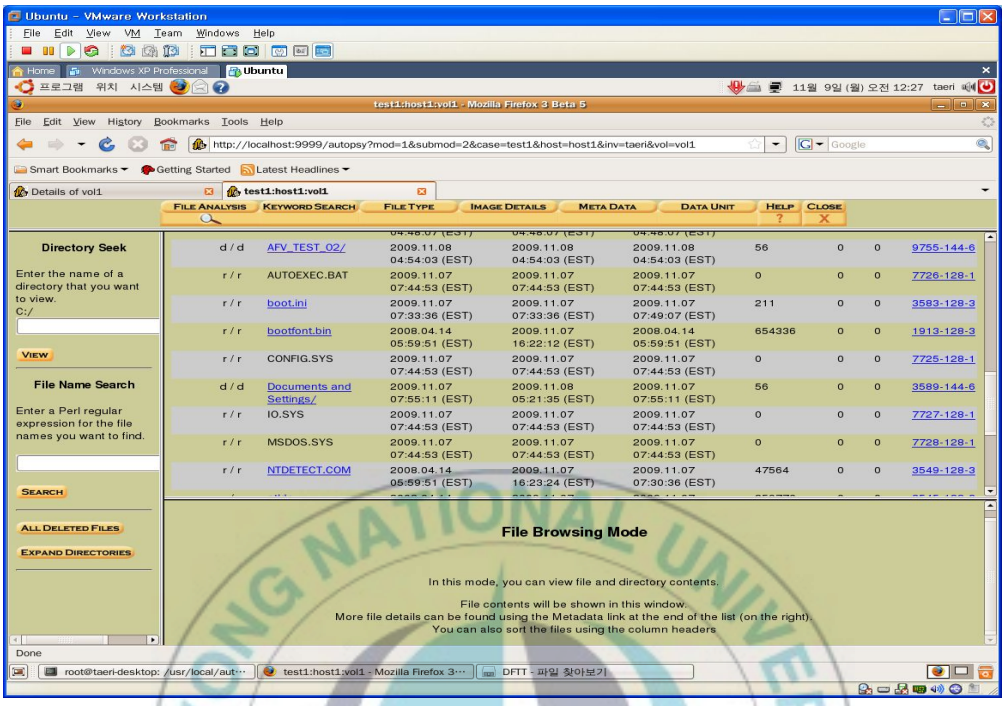
Test Cases	결 과
GV_TEST_01	불만족
GV_TEST_02	미 시행
GV_TEST_03	미 시행
DFRV_TEST_01	불만족
DFRV_TEST_02	불만족
FSV_TEST_01	만족
SSV_TEST_01	불만족
AFV_TEST_01	만족
AFV_TEST_02	만족

- GV_TEST_01 : 분석 작업 수행 결과를 <표 14>와 비교해 본 결과 이미지 파일 구조, 파일 시스템, 내부 구성 파일에 대한 상세 정보가 모두 일치하였으며, 수행한 모든 작업에 대한 이벤트 로그를 확인 가능하였다. 또한 동일한 분석 작업을 복수 수행한 결과가 모두 일치하였으며, 이미지 파일에 대한 MD5 값 변화가 발생하지 않았다. 하지만 file1.hwp 파일을 Microsoft Installer 로 인식하였으며, 한글 문자열 처리에서도 불완전한 결과를 보였다.
- GV_TEST_02 : 분석 대상을 이미지 파일만으로 제한하는 도구의 특성을 고려하여 테스트를 수행하지 않았으나, 도구의 오류 보고 확인을 위하여 다른 형태의 테스트를 설계하여 적용해 볼 수 있을 것이다.
- GV_TEST_03 : 분석 대상을 이미지 파일만으로 제한하는 도구의 특성을 고려하여 수행하지 않았다.
- DFRV_TEST_01 : <표 15>에 기재된 삭제된 파일 및 디렉터리 목

록의 내용 중 한글파일_in_deleted_dir2.hwp 파일에 대한 인식과 복구가 정확하게 이루어지지 않았다.

- DFRV_TEST_02 : <표 16>에 기재된 삭제된 파일들을 모두 식별하기 하였으나, 복구 후 단편화 된 파일 frag1.dat, frag2.dat, frag3.dat 의 MD5 값이 원본과 일치하지 않았다.
- FSV_TEST_01 : 분석 결과에 나타난 이미지 파일에 대한 파일 시스템 및 메타 데이터, 디렉터리 구조, 내부 구성파일 종류와 관련 정보들이 <표 17>과 모두 일치하였다.
- SSV_TEST_01 : UTF-8 인코딩 타입의 한글 문자열 외에는 전혀 알아볼 수 없는 형태로 처리하여, <표 18>에 나타난 모든 문자열들에 대한 누락 없는 탐색이 불가능하였다.
- AV_TEST_01 : 분석 결과에 나타난 두 파일에 대한 파일 생성 시간 정보가 <표 19>에 기재된 내용과 일치하였다.
- AV_TEST_02 : 해당 이미지 파일의 운영 체제, 시스템/ 사용자 정보, 설치된 프로그램들이 모두 테스트 설계와 일치하게 분석되었으며, 고유 정보가 변경된 파일 탐색 결과 또한 <표 20>의 내용과 일치하였다. 이미지 내부의 각 파일에 대한 해시 분석이 가능함을 확인하였다.

TSK & AutoPsy 의 경우 전반적으로 파일 타입 인식과 브라우징 측면에서 불만족스러운 결과가 많이 나타났으며, 특히 한글 관련 인코딩 타입 및 문자열에 대한 처리가 미흡하여 이에 대한 개선이 필요하다. 또한 단편화된 파일 처리 및 복구 기능에 대한 개선도 이루어져야 한다.



[그림 3] TSK & AutoPsy Test 화면

(3) ETRI Forensics

■ 시스템 정보

Intel(R) Core(TM)2 Quad CPU Q9300 2.50GHz, 3GB RAM
Microsoft Windows XP Professional Service Pack3

- GV_TEST_01 : 분석 작업 수행 결과를 <표 14>와 비교해 본 결과 이미지 파일 구조, 파일 시스템, 내부 구성 파일에 대한 상세 정보가 모두 일치하였으며, 문자열 및 파일 탐색 등 수행한 모든 작업에 대한 이벤트 로그를 확인 가능하였다. 또한 동일한 분석 작업을 복수 수행한 결과가 모두 일치하였으며, 이미지 파일에 대한 MD5 값 변화가 발생하지 않았다.

<표 23> ETRI Forensics Test 결과

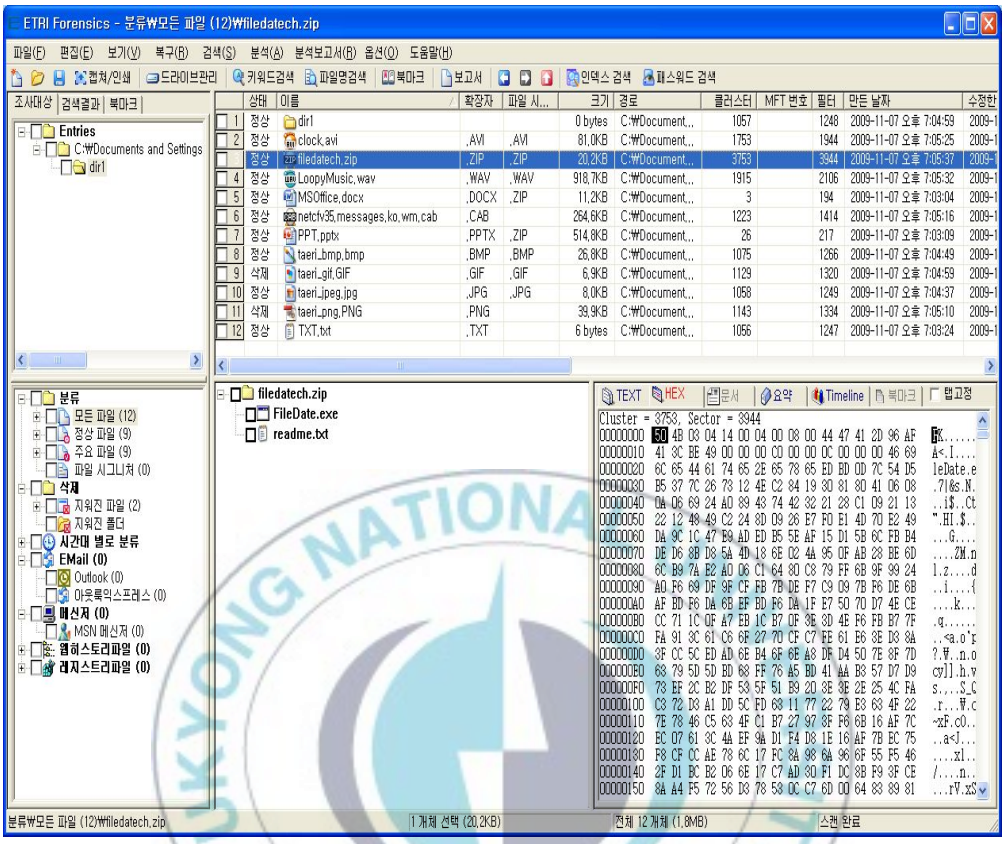
Test Cases	결 과
GV_TEST_01	만족
GV_TEST_02	만족
GV_TEST_03	만족
DFRV_TEST_01	만족
DFRV_TEST_02	불만족
FSV_TEST_01	만족
SSV_TEST_01	불만족
AFV_TEST_01	만족
AFV_TEST_02	만족

- GV_TEST_02 : 분석 작업 수행 시 고의로 제거된 외부 장치에 대한 오류 메시지를 정확하게 보고하였다. 하지만 도구가 최초 구동되는 시점에서 분석 시스템에 연결된 모든 장치에 대한 인식을 시도하는 형태로 동작하기 때문에 사용자 편의를 위하여 실시간으로 추가 또는 제거되는 외부 장치들에 대한 정확한 인식 여부가 가능하도록 보완될 필요가 있다고 사료된다.
- GV_TEST_03 : USB 디스크에 대한 분석 결과가 모두 일치하였으며. 분석 전 후의 MD5 값의 변화가 없으므로 쓰기방지장치 없이 연결되었음에도 데이터 변경이 이루어지지 않았음을 확인하였다.
- DFRV_TEST_01 : 도구가 식별한 삭제 파일 및 디렉터리 목록이 <표 15>에 기재된 내용과 모두 일치하였으며, 데이터 복구 후 MD5 값 비교를 통해 원본과 동일함을 확인하였다.
- DFRV_TEST_02 : <표 16>에 기재된 삭제된 파일들을 모두 식별하기 하였으나, 복구 후 단편화 된 파일 frag1.dat, frag2.dat, frag3.dat 의 MD5 값이 원본과 일치하지 않았다.
- FSV_TEST_01 : 분석 결과에 나타난 이미지 파일에 대한 파일 시

스템 및 메타 데이터, 디렉터리 구조, 내부 구성파일 종류와 관련 정보들이 <표 17>과 모두 일치하였다.

- SSV_TEST_01 : 문자열 탐색 질의를 통해 <표 18>에 나타난 대부분의 문자열들에 대한 탐색이 가능하였다. 하지만 한글 관련 문자열 탐색에서 UTF16BEHangul.dat 파일 내의 문자열 탐색에 실패하였으며, 키워드 검색 옵션 내에 존재하지 않는 인코딩 타입이므로 이에 대한 추가가 필요할 것이라 생각된다.
- AV_TEST_01 : 분석 결과에 나타난 두 파일에 대한 파일 생성 시간 정보가 <표 19>에 기재된 내용과 일치하였다.
- AV_TEST_02 : 해당 이미지 파일의 운영 체제, 시스템/ 사용자 정보, 설치된 프로그램들이 모두 테스트 설계와 일치하게 분석되었으며, 고유 정보가 변경된 파일 탐색 결과 또한 <표 20>의 내용과 일치하였다. 이미지 내부의 각 파일에 대한 해시 분석이 가능함을 확인하였다.

ETRI Forensics 의 경우 국내에서 개발 중인 도구답게 다른 도구들에 비해 국내에서 널리 사용되는 파일 타입에 대한 인식과 브라우징 측면에서 만족스러운 결과가 나타났으며, 특히 한글 관련 인코딩 타입 및 문자열에 대한 처리가 정확하게 이루어졌다. 하지만 문자열 검색과 관련하여 누락된 인코딩 타입에 대한 추가와 단편화된 파일 처리 및 복구 기능에 대한 개선이 이루어져야 한다.



[그림 4] ETRI Forensics Test 화면

VI. 결론

본 논문에서는 컴퓨터 관련 범죄의 급증에 따라 중요시 되고 있는 디지털 포렌식 도구들 중 증거 분석 기능에 초점을 두고 기존 도구들에 대한 분석을 바탕으로 디지털 증거 분석 도구 요구사항을 기술하였으며, 요구사항에 부합하여 대상 도구의 기능 및 동작, 분석 결과물에 대한 신뢰성 검증을 수행할 수 있도록 일반화된 테스트 절차를 제안하였다. 또한 실제 포렌식 도구인 EnCase 와 The Sleuth Kit & AutoPsy, ETRI Forensics 를 대상으로 제안된 테스트 절차를 활용하여 구체적인 모의 테스트를 설계하고 직접 수행하였다.

그 결과 3가지 도구 모두 많은 부분의 테스트를 성공적으로 통과한 반면, 단편화된 삭제 파일 복구 및 처리에 있어서 성능 개선이 필요함을 확인할 수 있었다. 또한 EnCase 와 The Sleuth Kit & AutoPsy 같은 국외 도구들은 국내에서 널리 사용되고 있는 파일 타입에 대한 인식과 한글 관련 문자열 처리가 미숙하여 이에 대한 보완이 필요함을 알 수 있었으며, ETRI Forensics 는 국내 개발 중인 도구인 만큼 이러한 부분에서 강점을 나타내었으나 한글과 관련하여 다양한 인코딩 타입을 추가적으로 고려해야 할 필요성이 있음을 발견하였다.

위와 같은 결과들을 통해 본 논문은 포렌식 관련 종사자 및 도구 개발자들에게 보다 많고 상세한 정보를 제공하며, 다양한 도구들을 대상으로 신뢰성 검증을 수행하고 평가가능하게 함으로써 도구 선택의 자유도를 높일 수 있을 것이다. 또한 국내 실정을 고려한 테스트 수행과 평가 기준 마련은 독자적인 디지털 포렌식 도구 개발과 관련 기술 발전에 이바지할 수 있을 것이라 사료된다. 이를 토대로 국외 포렌식 기술 의존으로 인한 국가적인 낭비를 줄여 새롭게 개발되는 미래의 포

렌식 도구들은 다양한 측면에서 국제적인 경쟁력을 지니며, 사이버 범죄를 처벌하고 예방하기 위한 핵심적인 도구로써 사용될 수 있을 것이다.

향후에는 보다 다양한 도구들을 대상으로 추가적인 테스트를 수행하면서 효율적인 테스트 설계 방법을 모색하고, 개별 생성한 이미지 파일들을 보완하여 상황에 따라 유용하게 적용 가능한 증거 이미지 파일 셋을 생성할 계획이다.

부 록

1. 테스트 이미지 생성 방법

다음은 본 문서 전반에 걸쳐 제시한 다양한 테스트 방법에 이용할 수 있는 이미지 파일 생성을 위한 일반적인 작업 절차이다. 이해를 돕기 위하여 디지털 분석 도구 기능 검증 항목 중 <표 11>문자열 탐색 기능 관련 테스트를 이미지 파일 생성 대상으로 하였으며, 작업 순서와 세부적인 사항 및 사용 툴들은 테스트 환경 및 설계자의 의도에 따라 다양하게 나타날 수 있다.

가. 테스트 상세 설계

특정 기능을 테스트하기 위해 구체적인 목표에 맞추어 전체적인 이미지 파일 구조 및 저장 데이터들을 설계하고, 생성 시 필요하게 될 작업 도구 및 환경과 사전 준비 파일들을 명시한다.

예) 다음 <표 22>는 검증 항목 SSV_03 증거 디스크 내 모든 영역에 대한 문자열 탐색이 가능 하다. 비할당 영역이나 단편화된 파일 또는 불연속적인 디스크 공간 내 문자열 탐색 등이 가능함을 의미한다. 를 테스트하기 위한 설계의 예이다.

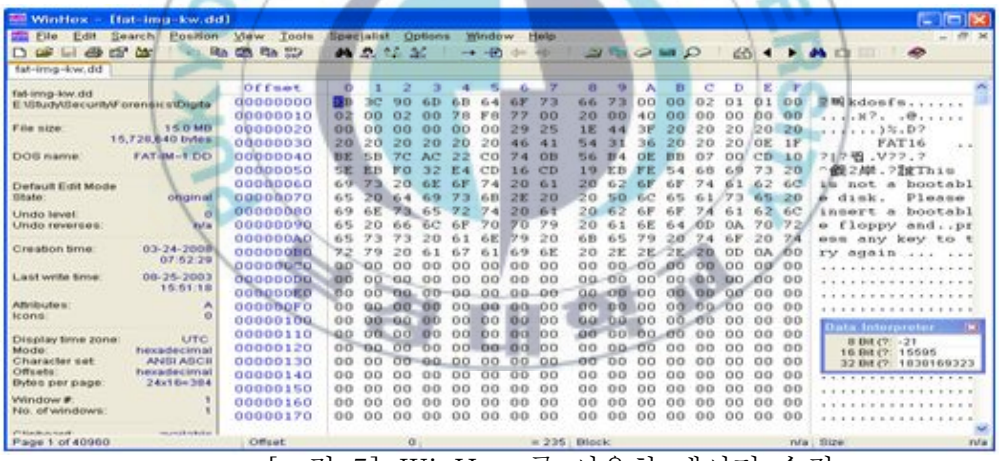
<표 24> 테스트 설계 표

번호	문자열	섹터	오프셋	문자열 저장 파일	설 명 (저장 위치 및 특징)
1	first	271	167	file1.dat	파일
2	second	272	288	file2.dat	파일
3	second	239	480	N/A	디렉터리 엔트리
4	1cross1	271	508	file1.dat & file2.dat	두 파일에 나누어진 문자열
5	2cross2	273	508	file3.dat	한 파일 내 연속적인 섹터

6	3cross3	283	508	N/A	비할당 영역
7	1slack1	272	396	file2.dat & file2 slack	한 파일 내 슬랙에 나뉨
8	2slack2	274	508	file3 slack & file4.dat	두 파일 슬랙에 나뉨
9	3slack3	277	385	file4 slack	슬랙 공간
10	fragment	278	507	file4.dat	단편화된 섹터
12	deleted	276	230	file5.dat	삭제된 파일

나. 사전 준비 파일 생성

상세 설계 단계에서 결정된 테스트 내용을 포함하며, 이미지에 복사될 파일들을 미리 생성한다. 일반적인 문서 편집용 도구를 제약 없이 사용 가능하며, 단편화된 데이터 삽입과 같은 특수한 수정을 요하는 파일의 경우 WinHex 등의 Hex Editor를 이용할 수 있다.



[그림 5] WinHex 를 이용한 데이터 수정

다. 이미지 파일 생성

테스트에 적합한 크기로 랜덤한 값을 가지는 이미지 파일을 생성한다.

예) Linux의 dd를 이용한 이미지 파일 생성 명령. 파일명은

fat-img.dd이며, 전체 파일 크기는 15MB이다.

dd if=/dev/random of=fat-img.dd bs=1m count=15

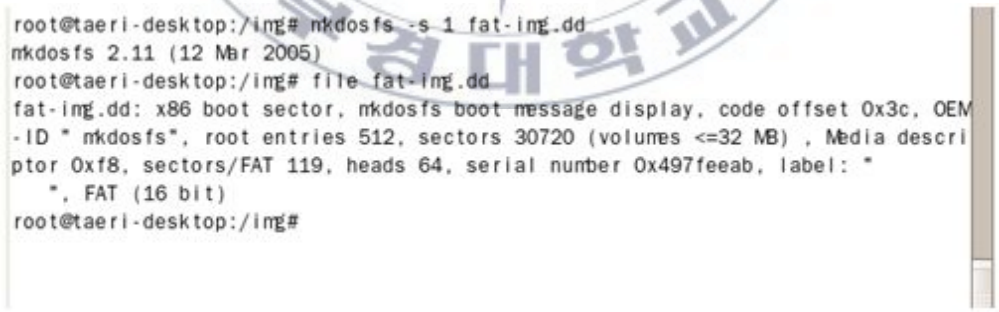
[그림 6] dd 를 이용한 이미지 파일 생성

라. 이미지 파일 포맷

테스트 설계에 해당하는 타입의 파일 시스템으로 앞서 생성한 이미지를 포맷한다.

예) FAT 파일 시스템 형태로 포맷하는 명령

mkdosfs -s 1 fat-img.dd



[그림 7] 이미지 파일 포맷 및 정보 출력 화면

마. 테스트 상세 설계

포맷한 이미지 파일의 조작을 위해 특정 위치로 마운트 시킨다.

예) Linux의 /mnt로 해당 이미지 파일을 마운트 시키는 명령
mount -o loop fat-img.dd /mnt

바. 파일 복사

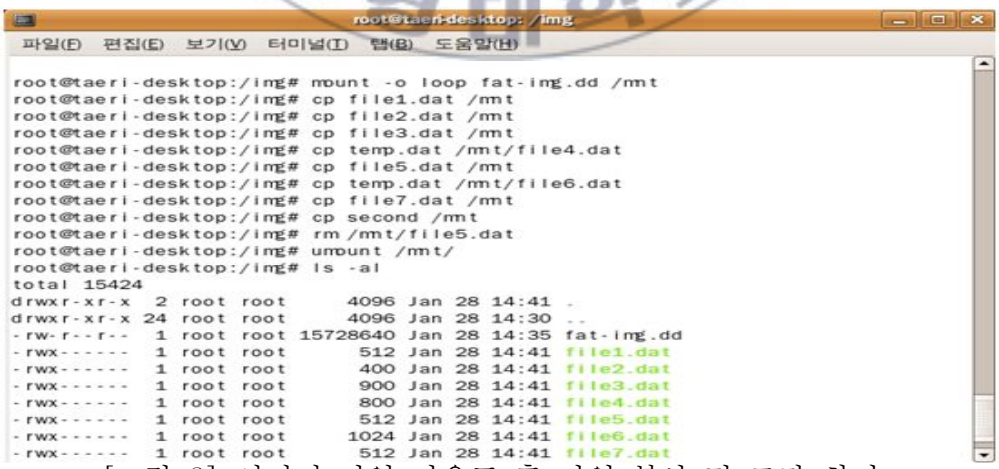
사전에 준비한 테스트 요소를 포함하고 있는 파일들을 마운트 시킨 이미지 파일로 복사한다. 테스트 설계에 따라 해당하는 파일들을 순서에 맞게 복사하고, 작업 내용을 명시한다.

예) /mnt에 마운트 된 이미지 파일에 사전 준비 파일을 복사하기 위한 명령
cp file1.dat /mnt
cp file2.dat /mnt

사. 이미지 파일 조작

특파일 삭제 등과 같이 설정된 테스트 내용에 맞게 이미지 내부 파일들을 조작한다.

예) 문자열이 포함된 파일 복사 후 삭제된 파일 상태로 만들기 위한 명령
rm /mnt/file5.dat



[그림 8] 이미지 파일 마운트 후 파일 복사 및 조작 화면

아. 이미지 파일 마운트 해제 및 추가 조작

파티션 테이블이나 디렉터리 엔트리 내용 수정과 같이 테스트 설계 사항 중, 파일 관련 이외의 것들에 대한 추가적인 이미지 파일 조작 과정이다. 테스트 목적과 이미지파일의 특징에 따라 데이터 조작을 위한 다양한 도구들이 사용 가능하며, 주로 Hex Editor를 이용한다.

자. 이미지 파일 생성 완료

테스트 설계의 모든 요소를 빠짐없이 적용 시킨 이미지 파일이 최종적으로 생성 완료되는 단계이다.

앞 서 설명된 일반적인 9단계 절차를 응용하여 디지털 증거 분석 도구의 성능 평가를 위한 각 검증 항목 별 다양한 테스트 설계 및 해당 증거 이미지 생성이 손쉽게 가능하다.

참고 문헌

[1] Amber Schroader, Tyler Cohen, "Alternate Data Storage Forensics", Syngress, 2007

[2] Albert J. Marcella, Robert S. Greenfield, "Cyber Forensics", CRC Press, 2002

[3] Debra Littlejohn, Shinder, “Scene of the Cybercrime (Computer Forensics Handbook)”, Syngress, 2002

[4] Anthony Reyes, Jack Wiles, “The Best Damn Cybercrime and Digital Forensics Book Period”, Syngress, 2007

[5] 손정환, 김귀남, “국내 디지털 포렌식 기술 현황과 발전 방안”, 한국사이버테러정보전학회논문지, 정보보증 논문지, 2005

[6] Computer Forensics Tool Testing(CFTT) Project, http://www.cftt.nist.gov/

[7] Computer Forensic Reference Data Sets (CFReDS) Project, http://www.cftt.nist.gov/

[8] 길연희, 홍도원, “디지털 포렌식 기술과 표준화 동향”, TTA Journal, IT Standard & Test, No.118, pp.75-81, 2008

[9] Guidance Software, EnCase, http://www.guidancesoftware.com/

[10] AccessData, Forensic Toolkit, http://www.accessdata.com/

[11] Internal Revenue Service, ILook Investigator, http://www.ilook-forensics.org/

[12] Technology Pathways, ProDiscover, http://www.techpathways.com/

[13] Armor Forensics, SafeBack, http://www.forensics-intl.com/

[14] ASR Data, SMART, http://www.asrdata.com/

[15] Vogon International, SDi32,
http://www.vogon-international.com/

[16] FInalData, Final Forensics, http://www.finaldata.com/

[17] A3Security, A3-AutoWatch, http://www.a3security,co,kr/

[18] Paraben, http://www.paraben-forensics.com/

[19] Forensic Acquisition Utilities,
http://users.erols.com/gmgarner/forensics/

[20] FTimes, http://ftimes.sourceforge.net/FTimes/

[21] liveview, http://liveview.sourceforge.net/

[22] Advanced Forensic Format Library, http://www.afflib.org/

[23] Automated Image and Restore(AIR),
http://air-imager.sourceforge.net/

[24] Rdd-2.0.7, http://sourceforge.net/projects/rdd/

[25] The Sleuth Kit(TSK), http://www.sleuthkit.org/sleuthkit/

[26] CDfs, http://users.elis.ugent.be/~mronsse/cdfs/

[27] CDrecord, http://freshmeat.net/projects/cdrecord/

[28] Autopsy Forensic Browser, http://www.sleuthkit.org/autopsy/

[29] File System Investigator, http://www.rossi.com/fstools/

[30] pyflag, http://www.pyflag.net/downloads/

[31] Event Log Parser, http://www.whitehats.ca/

[32] binutils, http://www.gnu.org/software/binutils/

[33] File AUdit Security Toolkit(FAUST),
http://security-labs.org/

[34] Forensic and Incident Response Environment(F.I.R.E),

http://biatchux.dmzs.com/

[35] Brian Carrier, "Open Source Digital Forensics Tools : The Legal Argument", @stake Research Report, 2002

[36] NIST CFTT, "Forensic String Searching Tool Requirements Specification", Public Draft 1 of Version 1.0, January 24, 2008

[37] NIST CFTT, "Deleted File Recovery Tool Specification", Draft for SC Review of Version 1.0, January 19, 2005

[38] Digital Forensic Tool Testing Images(DFTTI), http://dftt.sourceforge.net/

[39] National Institute of Justice, “Forensic Examination of Digital Evidence: A Guide for Law Enforcement”, NIJ Special Report, April 2004

[40] Brian Carrier, "Defining Digital Forensic Examination and Analysis Tools Using Abstraction Layers", International Journal of Digital Evidence, Winter 2003

[41] Ron Patton, "Software Testing second edition", Macmillan Computer Pub, 2005

[42] 김건우, 홍도원, “고속 디지털 포렌식 기술”, 정보보호학회지, pp.45, 2009

감사의 글

뒤를 돌아보며 살기에는 너무나도 치열하고 각박 하기만한 요즘 세상에 지난 2년간의 석사 과정은 제게 한없이 따듯하고 행복했던 추억이 아니었나 생각됩니다. 인생의 전환점이자 더 높은 곳으로 향하기 위해 박사과정으로의 진학을 결심한 이 때, 한 없이 부족하고 어리기만 했던 저를 가족 같은 아늑함과 보살핌으로 이 자리까지 올수 있도록 이끌어 주신 많은 분들께 새삼 더욱더 깊은 감사의 마음을 가지게 됩니다.

먼저 항상 자상한 배려와 세심한 보살핌으로 함께 고민해주시고 길을 열어주시는 지도교수 신상욱 교수님께 고개 숙여 감사드립니다. 앞으로도 많은 지도편달 부탁드리고, 더욱 열심히 하는 제자가 되겠습니다. 그리고 보다 좋은 논문 완성을 위해 단비 같은 조언과 아낌없이 격려해주신 이경현 교수님, 부족한 점을 채울 수 있도록 일깨워주신 신 원 교수님, 배움의 즐거움을 알 수 있게 해주신 조성진 교수님과 늘 학생들 곁에서 든든한 버팀목이 되어주시는 김창수 교수님, 그 외에도 훌륭한 가르침을 주신 여러 교수님들께 진심으로 감사드립니다.

그리고 누구보다 소중한 우리 LACUC 가족들, 편안함으로 보듬어 주시는 사모님, 큰 형처럼 감싸주신 태훈선배, 인기 많고 세련된 멋쟁이 진홍선배, 웃음이 끊이지 않게 해주셨던 재성선배, 연구실을 밝혀주는 우리의 에이스 꽃미녀 수완선배, 수많은 밤 희노애락을 함께 해준 태식선배와 물심양면으로 지원해 준 DH 장학재단 대표 도희선배 너무나 감사드립니다. 또 많은 시간을 함께 하게 될 우리 막내둥이들 주영이와 본민이, 격하게 아껴주고픈 학부생 애기들 모두 감사드리고, 모범이 되는 선배가 될 것을 감히 약속드립니다.

뼈 속부터 사나이 카리스마 아티스트 석철 형, 한 식구나 다름없는 LISIA의 서 철선배님과 영호선배님, 꿀맛 같은 휴식을 함께 해주는 영신이, 평생을 함께 하고픈 정이 넘치는 공간 W.A.P 선·후배님들, 언제나 저의 성공을 빌어주는 예비 아빠 재윤이와 예쁜 재수씨(애기 건강하게 낳

으시길..), 타지에서 고생하고 있는 우리 동기들 진영, 지현, 민수, 철호, 홍일점 현주, 대 끊긴 라인 후계자 석환이, 자랑스러운 남흔이. 태홍이 모두모두 깊은 감사드립니다.

마지막으로 너무나 크신 사랑과 변함없는 믿음으로 언제나 제 뜻을 지지해주시는 부모님, 장손 잘 되기만을 항상 기원해주시는 우리 식구들 감사드리고, 사랑합니다. E.C.H.O 가족들과 매력남 정민이형, 중동에서 구슬땀 흘리고 있을 성호, 뉴타입 카리스마 이전무 건욱, 신랄한 장사장 태수 & 부지런한 나경이 부부, 맛 집 전문가 도과장 경원, 중국인 다 된 창오, “준비 없는 이별” 오부장 주희, 고독한 멋쟁이 솔로 고래 보경, 코가 1cm만 높았어도 나라를 뒤엎었을 미남선생 종현 & 아리따운 재수씨, 믿음직한 송형과 알콩달콩 커플 귀염둥이 소영이, 영계킬러 신실한 말머리 성민, 믿음직한 친구 성수, 현호야 승혜야 내년엔 국수 먹자~, 예쁜이 삼총사 유부녀1호 착한 미지, 패셔니스타 민지, 노래 잘하는 센스쟁이 주윤이 모두 정말 고맙고, 앞으로 더 잘할게.

여러분 기대에 보답할 수 있도록 보다 멋진 제가 되겠습니다.